Volume II No. 6

(Issue 16)

£1

Unfortunately for the third issue running, the main part to my editorial has to be devoted to apologising for the delays. It is, regrettably, taking us rather longer than we had hoped to recover from the various problems of the summer. The last issue was delayed and this issue is yet again a "double" magazine.

Over the summer of 1983, we did have a number of problems, all of which have affected the magazine quite badly. For the time being, we are not putting a date on the magazine, simply the issue number. Until we can get completely back to normal we will be getting magazines out as soon as possible. We hope to be completely back to normal by the spring. The whole situation of membership at the moment has been confusing some members. At LASERBUG we base membership purely on issue numbers. If you joined LASERBUG from issue 13 for 6 months you will receive issues 13 to 18. Because issue 13 was in June, it does not mean as such that your membership expired in December, despite only receiving four magazines. We have not issued replacement membership cards but will do so on request. Once again, however, we must thank the large majority of members who have been so understanding. Finally on this topic I must make a point about the LASERBUG telephone number. The majority of LASERBUG is run from a private address, not an office. The telephone number is the number of the address, not LASERBUG. The owner of the address decided, for their own personal reasons, to change the telephone number – this was not our decision and basically had nothing whatsoever to do with us. The new LASERBUG telephone number will be made freely available to members on request. Please do not pester the printers of LASERBUG as they cannot help you.

Despite the delays with memberships, our ordering service is still working as normal – if you want backcopies, books or even printers, we will still be able to send them as usual. In fact, with printers, if we receive your order on a Monday, you will probably have your printer by the end of the week.

The LASERBUG 1983 Questionnaire has at this early stage provided some very interesting results. We have now stopped receiving questionnaires and the results will be printed around Easter, at the same time as the LASERBUG Awards are announced.

The LASERSOFT software service will now be launched in the spring – when it starts up, software available will include Games, Utilities, Educational Programs and possibly even Business Programs. Full details will be supplied on receipt of an SAE.

We have now added further printers to our range and lowered the price on them all, so we hope they will be of even more interest now. Also our supplier now has available monitors and even 3" disk drives!

Finally, I must thank you all, on the whole, for being so patient. Hopefully the next issue shouldn't be too far away. LASERBUG is still in business and we intend to keep it that way.

Paul Barbour

contents

- Editorial 2
- News 2
- Meeting Place 3
- Local Clubs 3
- Specialist Clubs 5
- ROM Scanner 5
- A.D.C. Test Tool 6
- Variable Dump 8
- Colour on view 8
- Character Designer 9
- Simple Sprites 13
- R.H. Light Pen 14
- Easy Speech 15
- Advertisements 16
- Introducing the electron 18
- Micro Writing 18
- I.C.B.M. Attack 19
- Problems between BBC and Electron 21
- Using Atari Style Joysticks 21
- Screen Dump for White Knight 22
- Cassette Recorder Review 23
- Music Editor Review 24
- Verity 24
- Micro Writer Mouse 25
- BBC Micro V Electron 26
- Credits 27

NO PHOTOS

We apologise this week for the lack of photos in the news section. This will be back to normal next issue. . .

ACORN AND EDUCATION

Acornsoft in conjunction with Applied Systems Knowledge Ltd. have launched a new range of educational software for the BBC Micro and Electron. The programs cover numeracy and literacy and are aimed at the 3–11 age group. They are all entertaining and non-violent and are carefully written – a child is never pressured or put down if he gives a wrong answer. The programs are Number Chaser; Words, Words, Words; Cranky; Table Adventures; Children From Space; Facemaker; Hide and Seek; Let's Count; Number Gulper and Number Puzzle. From the little we have seen of the programs, they look extremely good. Acornsoft Ltd., 4a Market Hill, Cambridge, CB2 3NJ.

ADVENT

Micro-Advent, producers of the successful Microvoc sound system, have launched a new range of software which is "enhanced by use of the Microvoc sound system". The first is Music Tools which consists of five programs – an organ program, an auto tune generator, a symbol writer, a sound envelope explorer and a three part tune player. The second program is "Pieman" which is a more complex Simon type game. Musictools costs £5.75 and Pieman £3.75. Micro-Advent, Ashlyn House, 113 Writtle Road, Chelmsford, Essex.

40/80 TRACK SOFTWARE

One of the problems in buying disk drives is whether to buy 40 or 80 track. In the end, a lot of people end up buying 40/80 switchable drives, which is more expensive. Acornsoft, realising this, have launched a unique dual format disk system that allows a disk to be read by both 40 and 80 track drives. All newly published titles will be released on the new style disks, and older titles sold on the normal disks will be made available in the new format later this year. The new disks will cost the same as the old ones.

ZYGON STAND

One of the problems with computer tables and stands is that they soon run out of room as you extend your computer. Zygon Products have now added to their range of stands the Zygon Delta as their top model. As you can see, it has room for every peripheral you would want. The Zygon range starts at £59 + VAT and goes up to their top model for £96 + VAT. Zygon Products, 9 Sherdes Drive, Hoddesdon, Hertfordshire, EN11 8LH.

PENGUIN/ACORN LIBRARY

Penguin and Acorn, "two of the mightiest names in the leisure industry", are to produce a series of books and book/software combinations entitled 'The Penguin Acorn Computer Library'. To mark the recent launch of the Electron, the first titles will be 'The Acorn Guide To The Electron' and 'Games And Other Programs For The Acorn Electron'. At least eight titles will be published in the first year, each one selected jointly by Acorn and Penguin. Future titles will be 'How To Write Arcade Games' by Jonathan Griffiths (author of 'Snapper and Planetoifd') and 'How To Write Adventure Games' by Peter Killworth.

INTEGREX COLOUR

The two main colour printers used with the BBC Micro at the moment are the CGP-115 4 colour plotter and the Seikosha GP-700A (both available from LASERBUG – see the back cover). Now Integrex have launched the BX-80 7 colour printer designed especially for use with the BBC Micro. It connects via the RS423 (?) and has a 3.5k buffer. It can print at 125 CPS and works on a 5 × 7 matrix. A screen dump listing is supplied for all BBC modes including 7. Two printout size options are available for MODE 7 including the ability to print two pages side by side. Integrex Ltd., Church Gresley, Burton-on-Trent, Staffs., DE11 9PT. (For American members they have an office in Philadelphia – Tel. No: 215 6 27 09 66).

PRODUCTION TO DATE

By the end of last year, 225 000 BBC Micros will have been made – total sales thus between £80 and £90m!

SUPERIOR DECISION AND NEW RELEASES

Early this year there was a threatened legal battle between Century Electronics, Superior Software and Ocean. Century produce the arcade game 'Hunchback' and both Superior and Ocean produced versions for home computers. Now, Century has granted Superior Software a license to sell 'Hunchback' for the BBC, and Ocean to produce it for other computers including the Spectrum and Electron (although no Electron version is available yet). The decision to give Ocean the license for the Electron version seems rather silly to us as Superior's BBC version could be easily converted for the Electron – however, ours is not to reason why. At the same time, Superior announced the launch of 6 new programs – '2002' (a docking game), 'Star Trek Adventure', 'Lost City', 'Gideon's Gamble', 'Fairground' (Carousel), and 'Crazy Painter' (Amidar). All programs cost £7.95 inclusive. Superior Software, 69 Leeds Road, Bramhope, Leeds.

ACORN MARRIAGE GUIDANCE!

By the release of two computer programs, 'I DO' and 'The Dating Game', Acorn expect to "improve people's love lives and help bring down the divorce rate"! The programs use personality questionnaires developed by Professor Hans Eysenck and Dr. Glenn Wilson. 'I Do' was developed to help couples understand their individual personalities and their relationships. More than 400 multiple choice questions built into the program test the most important aspects of a person's character and attitudes in such areas as aggressiveness, marital satisfaction, political stance, sex drive and sexual satisfaction. By matching the answers for each partner in a couple, the program can point out where a couple are in disagreement and those where they are in harmony.

'The Dating Game' analyses an individual's character and personality and provides hints to help him or her find compatible friends or lovers. The package consists of 4 programs, the most powerful of which is 'The Dating Game' itself. This can match up to 40 people at one time and rank the others in descending order of compatibility. Compatibility ratings for both friendships and romantic attachments are catered for and the program also works for homosexual couples. The second program is 'Love Style' which allows the computer to work out a personal Love Style profile. 'Preferred Relationship' attempts to indicate the kind of person you should be looking for in a serious relationship. 'Dating Skills' examines, in a humorous way, a person's social skills in dating and mating. The programs are available for both the Electron and the BBC Micro for £12.65.

MORE MICRO POWER POWER

Micro Power have been extremely busy – not only have they released versions of most of their BBC software for the Electron, but have also launched a number of new titles. Old BBC programs now available for the Electron are 'Escape From Moonbase Alpha', 'Moonraider', 'Swoop', 'Felix And The Fruit Monsters', Bandits At 3 O'Clock', 'Croaker', 'Chess', 'Killer Gorilla' and 'Felix in the Factory'. New programs (for the BBC unless otherwise stated) are 'Nemesis, 'Space Jailer', 'Martian Attack', 'Poistron' (and Electron), 'Zarm', 'Intergalactic Trader' (Electron), 'Cosmic Combbat', 'Cybertron Mission' (Electron), and 'Bumble Bee'. The majority of programs for the BBC Micro are to be converted shortly onto the Electron. Micro Power, Northwood House, North Street, Leeds, LS7 2AA.

LATE NEWS

Obviously, with the delay in producing this issue, some of the above news may be out of date. Completely up to date news next issue. . .

meeting place

It is clear from the early results in the questionnaire that 'Contacts' is no longer popular and hence we have decided to withdraw that particular feature. Instead we will be printing 'Meeting Place' every month. How about a few more clubs? If you run a club don't forget to drop us details for free publicity!

local clubs

ABERDEEN

Contact: Jack Lavety
Address: 16 Deeside Park, Aberdeen, AB1 7PO
Club Name: Grampian Amateur Computer Society
Meeting Place: Thistle Lane, Aberdeen.

BARNSLEY

Contact: James Bridson
Address: 39 Keresforth Hall Road, Kingstone, Barnsley, S. Yorkshire, S70 6NF.
Telephone: 0226 41753 (after 4.30 p.m. please)
Club Name: Barnsley Computer Users Group

BOLTON

Contact: Chris Shee
Telephone: 0942 720984
Club Name: Bolton BBC Micro and Electron User Group
Meeting Place: Variable
Times: Alternate Tuesdays

BRIGHTON & HOVE

Contact: I. Smith
Address: 30 Leicester Villas, Hove, E. Sussex, BN3 5SQ
Club Name: Brighton, Hove and District Computer Club
Meeting Place: Southwick Community Centre
Times: Every second Wednesday between 7.30 and 10.00 p.m.

CARDIFF

Contact: Geoff Barker
Telephone: Penarth 701023
Club Name: Cardiff BBC Computer Club (CBCC)
Meeting Place: Applied Science Lecture Theatre of University College, Newport Road, Cardiff.
Times: Alternate Wednesday evenings
Other Details: Extensive facilities at Lecture Theatre. Had 60 members after only 3 months.

CHELMSFORD

Contact: G.W. Goodacre
Address: 34 Quilp Drive, Chelmsford, CM1 4YA
Club Name: Chelmerbug
Meeting Place: Local school
Times: First Wednesday of every month

CROYDON

Contact: I.M. Khabaza
Address: 10 Lawrence Road, South Norwood, London, SE25 5AA
Telephone: 01 653 3207/01 653 4610
Club Name: Croydon Computer Club BBC Group
Meeting Place: Croydon Central Library
Times: 1st and 4th Tuesdays of each month at 7 p.m.

GREAT YARMOUTH

Contact: Nick Lamb
Address: 23 Gaywood Close, Caistor-on-Sea, Great Yarmouth, Norfolk, NR30 5RD
Telephone: 0493 728442

ISLE OF SKYE

Contact: C.J. Manvell
Address: Tigh na Pairc, 25 Breacais Iosal, Isle of Skye, IV42 8QA
Club Name: Skye and Lochalsh Computing Society
Other Details: Caters for all machines with BBC Micro dominating

LONDON (N11)

Contact: John Claydon
Telephone: 01 889 5446
Club Name: North London BBC Microcomputer Users Group and Education Workshop
Meeting Place: Bounds Green Junior School, Park Road, N11
Times: Second Sunday of each month at 2.00 p.m.
Other Details: Fee of approx. £1 per meeting to cover costs

LONDON (W)

Contact: Chris Drage
Telephone: 01 743 1579 (evenings)
Club Name: West London BBC Micro Users Sub-Group
Meeting Place: Back room of the Fox and Goose Pub, Hanger Lane, Alperton
Times: 3rd Tuesday of each month at 7.30 p.m.
Other Details: Membership £6.00 per annum (£3.00 to under 16's and O.A.P.s)

NOTTINGHAM

Contact: John Day
Presetel Mailbox: 602 225 660
Club Name: Nottingham BBC Micro Users Group
Meeting Place: Congregational Federation Centre, Castle Gate
Times: 7.30 p.m.

ORPINGTON

Contact: Norman Lambert
Address: 11 Vinson Close, Oprington, Kent, BR6 0EQ
Club Name: Orpington Computer Club
Meeting Place: Local Church Hall
Times: Every Friday evening

SHEFFIELD

Contact: John Fryer
Address: 17 Edgedale Road, Sheffield, S7 2BQ.
Club Name: ABUG (Sheffieldd Acorn & BBC Microcomputers User Group)
Times: First and third Wednesday of each month
Other Details: Has a mixed membership of around 70 people with a variety of abilities and interests.

SOUTHPORT

Contact: Ray Mitcham
Address: 5 Easedale Ave., Ainsdale, Southport
Telephone: 79936
Club Name: Southport BBC Micro Group

SWANSEA

Contact: Nick Goodwin
Club Name: South West Wales BBC and Electron User Group (allied with the Swansea Computer Club)
Meeting Place: Above the Three Lamps Pub in Swansea
Times: Every Tuesday

SWEDEN

Contact: Janne Soderberg
Address: Frihetsvagen 32, S-175 33 Jarfalla, Sweden
Telephone: 0758 31753
Other Details: Caters for the Atom as well as the BBC Micro

WAKEFIELD

Contact: Richard Sterry
Address: 1 Wavell Garth, Sandal, Wakefield, West Yorkshire, WF2 6JP
Telephone: Wakefield 25515

WICKFORD

Contact: Ms. J. Lines (or Mr. F. Lines)
Address: 97 Oakhurst Drive, Wickford, Essex, SS12 0NW
Telephone: Wickford 63396
Club Name: SEEBUG (South East Essex BBC User Group)
Meeting Place: Billericay School, Billericay
Times: Second Tuesday of each month between 7.30 and 9.30 in the Computer Room.

WREXHAM

Contact: Mike Houghton
Address: 1 Sherwell Avenue, Wrexham, Clwyd, LL13 9TZ
Club Name: Wrexham & District Computer Club
Times: Every Thursday

8-DFS

8-INCH FLOPPY DISK FILING SYSTEM

FOR THE BBC!

☆ **Fully Compatible with ACORN DFS**

☆ **Approved by ACORN COMPUTERS Ltd.**

☆ **Nearly 1.2 Mega-bytes of on-line Storage**

☆ **8 User Directories – 248 Files!**

8-DFS ROM Complete With Installation Manual

£44.95 + 1.50 p&p

Manual Available Separately £3.00

VOGAN Products, The White House, 21 Grove Road,

Hazlemere, Bucks., HP15 7QY

DISKS

Contact: Peter Hughes
Address: The Lending Library, Five Marsh Street, Bristol, BS1 4AA
Club Name: Format 40/80 Disk Club
Other Details: The idea of the Club is to "exchange programs of all kinds including school, educational and of course games. This will be done by a monthly disk to members ready to BOOT up with menu".

MICRONET

Contact: George H. Foot
Club Name: Micronet Independent User Group
Other Details: Can be found on Presetel page 8008100

NETWORKS

Contact: Tom Short or Mike Taylor
Address: The Computer Centre, Bedford College of Higher Education, (Mander) Cauldwell Street, Bedford, MK42 9AH
Club Name: BBC Network User Group (?)
Other Details: Set up to cater for BBC Micro networks of any description. Their main aims are (i) to pool expertise and experiences into a central area and then be able to communicate the information through meetings and a newsletter/magazine, and (ii) to act as a more effective pressure group or channel through which to pass on complaints to manufacturers.

rom scanner

There are spaces on the BBC Micro for 4 sideways ROMs – BASIC is one of these, a DFS would be another, a wordprocessor might be a third, which leaves one socket, perhaps for FORTH or BCPL? Some people find that four ROMs isn't enough and so fix a ROM extension board, increasing the number of ROMs that can be fitted to 16.

Without unscrewing your computer, wouldn't it be nice to have a program that lists out all the ROMs fitted inside your computer? *HELP does this of course but the program below will list out for all ROMs fitted (i) each ROMs number, (ii) how you can disable the ROM, (iii) the ROMs name, (iv) the ROMs version, (v) the ROMs copyright, (vi) whether it has a service entry and (vii) whether it has a language entry. If you know enough about the format of ROMs, the program is easily extendable.

If you do disable a ROM (the program tells you how), then it will not be present until the next BREAK. Disabling a ROM can be used to temporarily fool the system into thinking it isn't fitted – some ROMs interfere with each other and certain functions of the computer you will find, i.e. Disk Doctor & View, Acorn Prestel ROM and disk formatting, etc.

```
10 REM    ROM-Scanner
20 REM  by Paul Barbour
30 :
40 REM    Version 1.0
50 :
60 REM  BBC Micro Only
70 REM   OS 1.20 Only
80 :
90 REM (c) LASERBUG 1984
100 :
110    :::::
120 :
130 MODE7
140 VDU23,1,0;0;0;0;
150 PROCexpansion
160 PROCread
170 END
180 :
190    :::::
200 :
210 DEFPROCexpansion
220 PRINTCHR$134;"Is a ROM expansion board fitted on
this";
230 PRINTCHR$134;"computer ? ";CHR$130;"(Yes/No/Don't
know)"
240 REPEATanswer$=GET$:UNTILINSTR("YNDynd",answer$)
250 IFanswer$="N"ORanswer$="n"THENmax%=12:ELSEmax%=0
260 ENDPROC
270 :
280 DEFPROCread
290 FORrom%=15TOmax%STEP-1
300   CLS
310   IF?(&2A1+rom%)=0THENPRINTCHR$129;"ROM ";rom%;"
(&";~rom%;") Not Present":GOTO460
320   PRINTCHR$134;"ROM Number:"'CHR$130;rom%;" (&";~
rom%;")"
330   PRINTCHR$134;"ROM Disabled By:"'CHR$130;"?";&2A
1+rom%;"=0 (?&";~&2A1+rom%;"=0)"
340   PRINTCHR$134;"ROM Name:"'CHR$130;
350   mem%=9:PROCprint
360   PRINTCHR$134;"ROM Version:"'CHR$130;
370   PROCprint
380   PRINTCHR$134;"ROM Copyright:"'CHR$130;
390   mem%=FNread(7)+1:PROCprint
400   PRINTCHR$134;"ROM Type:"
410   val%=rom%?&2A1
420   PRINTCHR$130;"Service Entry -";CHR$133;
430   IFval%>128THENPRINT"Yes":val%=val%-128:ELSEPRIN
T"No"
440   PRINTCHR$130;"Language Entry -";CHR$133;
450   IFval%>64THENPRINT"Yes":val%=val%-64:ELSEPRINT"
No"
460   PRINT'''CHR$131;"Press SPACE"
470   REPEATA=GET:UNTILA=32
480   NEXT
490 END
500 :
510 DEFFNread(loc%)
520 loc%=loc%+&8000
530 Y%=rom%
540 ?&F6=loc%MOD256
550 ?&F7=loc%DIV256
560 =USR(&FFB9)MOD256
570 :
580 DEFPROCprint
590 REPEAT
600   char%=FNread(mem%)
610   IFchar%>31ANDchar%<128THENVDUchar%
620   mem%=mem%+1
630   UNTILchar%=0
640 PRINT
650 ENDPROC
```

6 adc test tool

This program will prove invaluable for anyone working on the BBC Micro's Analogue-Digital Convertor.

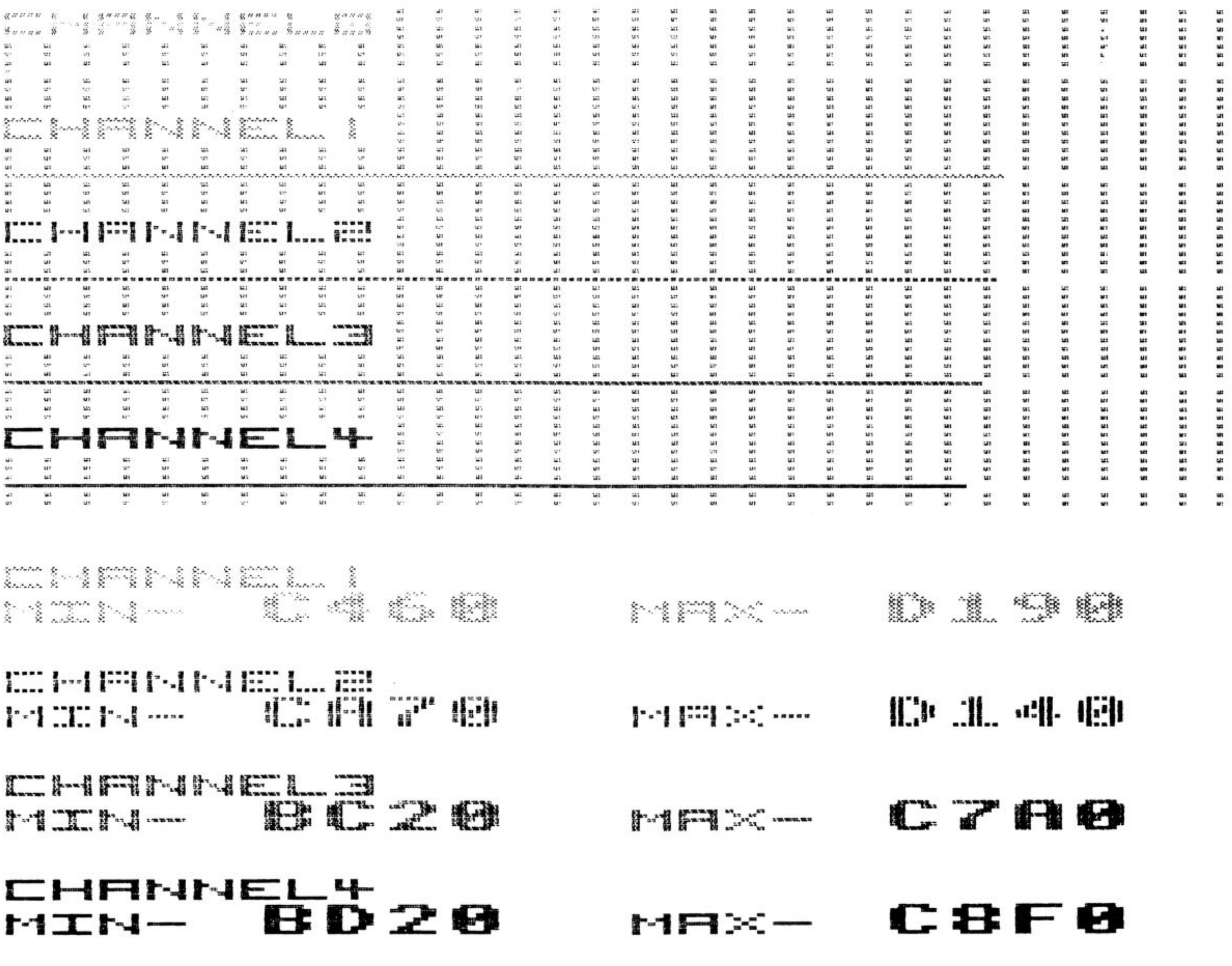
Most people, if they want a quick guide to the value on any one channel, type in a line such as REPEAT:PRINTADVAL(1):UNTILO. It works but isn't quite the best method around.

The program below will display graphically the value of all five channels. The four main channels have been scaled down to an accuracy of 8 bits. At the bottom is a note of the maximum and minimum values received so far on channels 1–4. These values are pure ADVAL readings (in hex). If at any time you want to re-measure the max/min, pressing R will reset them to their default values.

If you make any improvements to the program, such as speeding up the reading speed, please let us know.

```
1180 IFX%<min2%THENmin2%=X%
1190 X%=ADVAL(3)
1200 IFX%>max3%THENmax3%=X%
1210 IFX%<min3%THENmin3%=X%
1220 X%=ADVAL(4)
1230 IFX%>max4%THENmax4%=X%
1240 IFX%<min4%THENmin4%=X%
1250 COLOUR3
1260 PRINTTAB(4,17);"min1%;"  ";TAB(14,17);"max1%;"  "
1270 COLOUR5
1280 PRINTTAB(4,20);"min2%;"  ";TAB(14,20);"max2%;"  "
1290 COLOUR6
1300 PRINTTAB(4,23);"min3%;"  ";TAB(14,23);"max3%;"  "
1310 COLOUR7
1320 PRINTTAB(4,26);"min4%;"  ";TAB(14,26);"max4%;"  "
1330 ENDPROC
1340 :
1350 DEFPROCframe
1360 COLOUR2
1370 PRINTTAB(0,0);channel0$
1380 COLOUR3
1390 PRINTTAB(0,3);channel1$
1400 COLOUR5
1410 PRINTTAB(0,6);channel2$
1420 COLOUR6
1430 PRINTTAB(0,9);channel3$
1440 COLOUR7
1450 PRINTTAB(0,12);channel4$
1460 COLOUR3
1470 PRINTTAB(0,16);channel1$
1480 PRINTTAB(0,17);min$;TAB(10,17);max$
```

Continued on page 23



```
>L.
   10 REM    ADC TESTER
   20 REM  by Paul Barbour
   30 :
   40 REM    Version 1.0
   50 :
   60 REM  BBC Micro Only
   65 REM   OS 1.20 Only
   70 :
   80 REM (c) LASERBUG 1984
   90 :
  100    :::::
  110 :
  120 MODE2
  130 VDU23,1,0;0;0;0;
  140 PROCset_up
  150 PROCframe
  160 REPEAT
  170   PROCchannel0
  180   PROCchannel1
  190   PROCchannel2
  200   PROCchannel3
  210   PROCchannel4
  220   PROCmaxmin
  230   IFINKEY(-52)THENmin1%=&FFFF:max1%=0:min2%=&FFFF
:max2%=0:min3%=&FFFF:max3%=0:min4%=&FFFF:max4%=0
  240   UNTIL0
  250 END
  260 :
  270    :::::
  280 :
  290 DEFPROCchannel0
  300 COLOUR2
  310 X%=(ADVAL(0)AND3)*426.666667
  320 MOVE0,960:PLOT7,1280,960
  330 GCOL0,2
  340 MOVE0,960:DRAWX%,960
  350 ENDPROC
  360 :
  370 DEFPROCchannel1
  380 COLOUR3
  390 X%=FNadval(1)
  400 MOVE0,864:PLOT7,1280,864
  410 GCOL0,3
  420 MOVE0,864:DRAWX%,864
  430 ENDPROC
  440 :
  450 DEFPROCchannel2
  460 COLOUR5
  470 X%=FNadval(2)
  480 MOVE0,768:PLOT7,1280,768
  490 GCOL0,5
  500 MOVE0,768:DRAWX%,768
  510 ENDPROC
  520 :
  530 DEFPROCchannel3
  540 COLOUR6
  550 X%=FNadval(3)
  560 MOVE0,672:PLOT7,1280,672
  570 GCOL0,6
  580 MOVE0,672:DRAWX%,672
  590 ENDPROC
  600 :
  610 DEFPROCchannel4
  620 COLOUR7
  630 X%=FNadval(4)
  640 MOVE0,576:PLOT7,1280,576
  650 GCOL0,7
  660 MOVE0,576:DRAWX%,576
  670 ENDPROC
  680 DEFPROCgrid
  690 GCOL0,4
  700 FORX%=0TO1280STEP40
  710   MOVEX%,1024:PLOT21,X%,560
  720   NEXT
  730 ENDPROC
  740 :
  750 DEFFNadval(A%)
  760 =(ADVAL(A%)DIV255)*Z%
  770 :
  780 DEFPROCset_up
  790 *FX189,8
  800 PROCgrid
  810 Z%=5.01960784
  820 min1%=&FFFF:max1%=0
  830 min2%=&FFFF:max2%=0
  840 min3%=&FFFF:max3%=0
  850 min4%=&FFFF:max4%=0
  860 VDU23,128,0,0,250,130,131,130,250,0
  870 VDU23,129,0,0,47,40,239,40,40,0
  880 VDU23,130,0,0,162,178,170,166,162,0
  890 VDU23,131,0,0,139,202,171,154,139,0
  900 VDU23,132,0,0,232,8,136,8,239,0
  910 VDU23,133,0,0,62,34,42,34,190,0
  920 VDU23,134,0,0,8,8,8,8,136,0
  930 VDU23,135,0,0,62,2,62,32,190,0
  940 VDU23,136,0,0,62,2,14,2,190,0
  950 VDU23,137,0,0,40,40,62,8,136,0
  960 VDU23,138,0,0,139,218,171,138,138,0
  970 VDU23,139,0,0,232,37,226,37,40,0
  980 VDU23,140,0,0,128,0,62,0,128,0
  990 VDU23,141,0,0,139,216,168,136,139,0
 1000 VDU23,142,0,0,232,140,138,137,232,0
 1010 VDU23,143,0,0,128,128,190,128,128,0
 1020 channel$=CHR$128+CHR$129+CHR$130+CHR$131+CHR$132
 1030 channel0$=channel$+CHR$133
 1040 channel1$=channel$+CHR$134
 1050 channel2$=channel$+CHR$135
 1060 channel3$=channel$+CHR$136
 1070 channel4$=channel$+CHR$137
 1080 max$=CHR$138+CHR$139+CHR$140
 1090 min$=CHR$141+CHR$142+CHR$143
 1100 ENDPROC
 1110 :
 1120 DEFPROCmaxmin
 1130 X%=ADVAL(1)
 1140 IFX%>max1%THENmax1%=X%
 1150 IFX%<min1%THENmin1%=X%
 1160 X%=ADVAL(2)
 1170 IFX%>max2%THENmax2%=X%
```

8 colour on view

For most work people do using VIEW, they will find that MODE(3) is the easiest to use because of its 80 column screen. However, unless you are using a high resolution monitor, which can cost over £500 for an RGB one, you will probably find the white text on black background rather hard to use over long periods. Also in MODEs 3 and 6, you do not get the benefit of the lined screen. The answer, of course, is to change the foreground and background colours – as you would do in BASIC with VDU19.

Most people's first thought is that VIEW does not have a VDU command or anything similar and so you can't change colours. This, I am afraid, shows a lack of understanding of the BBC Micro. The VDU command in BASIC (or the PRINT command) of course passes a variable to the VDU drivers inside the BBC Micro – the third way of doing this is with the ConTRoL key.

In BASIC, type:

```
CLS
```

and the screen will clear. What this command actually does is send the number 12 to the VDU drivers. Try entering:

```
PRINT CHR$(12)
```

and you will find that the screen clears again. Use VDU:

```
VDU 12
```

and the screen clears once more. However, the fourth way to clear the screen is to press:

```
CTRL–L
```

(hold the CTRL key down as if it was a SHIFT key and press the letter L). The L gives the value 12 to the drivers and the screen clears. How? If you type:

```
A=GET
< Press CTRL–L >
PRINT A
```

you will find that A is 12. The normal value of the L key is 76. Pressing the CTRL key subtracts 64 from that value, giving 12.

So how does this help us? Well, suppose we want to change the background in VIEW from black to blue. Using a VDU command we would have to use:

```
VDU19,0,4,0,0,0
```

19 is the letter S (PRINTCHR$19+64), 0 you will find is @ and 4 is D. So, in VIEW press:

```
CTRL–S
CTRL–@
CTRL–D
CTRL–@
CTRL–@
CTRL–@
```

and the background should change (remember that CTRL is the control key and that you should be in a mode other than 7).

Use CTRL–S/CTRL–@ to define the background colour and CTRL–S/CTRL–A for the foreground colour (in MODEs 0, 3, 4 and 6). You will find the following table helpful in doing this:

```
@ – 0 – (Black)
A – 1 (Red)
A – 2 (Green)
A – 3 (Yellow)
A – 4 (Blue)
A – 5 (Magenta)
A – 6 (Cyan)
A – 7 (White)
```

This applies both to the VIEW wordprocessor and VIEW-SHEET.

variable dump

When trying to edit a program, one of the very helpful functions it would be helpful to have in BASIC is a VLIST command, i.e. a command that lists all the variables in a program. Unfortunately BBC BASIC doesn't have this facility, so instead below we present our variable dump procedure.

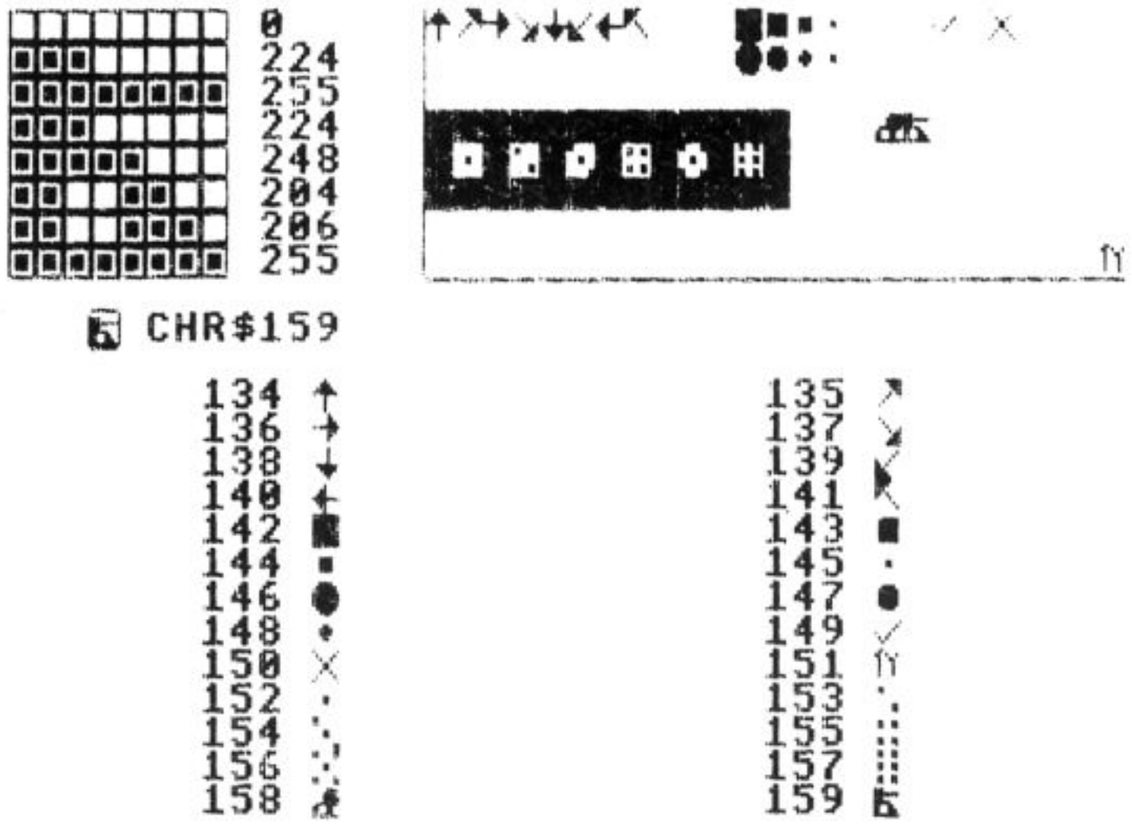
To use the program, type it in and save it using *SPOOL (i.e. *SPOOL VARDUMP/LIST/*SPOOL) and then later add it onto the program you are trying to edit using *EXEC VARDUMP. When called using PROClist_var it will list out the names and current values of all variables and will print the maximum size of all arrays. This should prove extremely useful to all programmers.

Dan O'Brien promises other good programs in the future, including a very special disk compactor. . .

```
/L.
30000 REM     Variable Dump
30010 REM    by Dan O'Brien
30020 :
30030 REM      Version 1.0
30040 :
30050 REM BBC Micro & Electron
30060 :
30070 REM   (c) LASERBUG 1984
30080 :
30090    :::::
30100 :
30110 DEFPROClist_var
30120 PRINT"Press (CTRL-B) to dump to printer,"'"any ot
her letter to dump to screen";GET$;
30130 LOCALA%,E%,I%,J%,L%,K%,M%
30140 FORA%=65TO122
30150   PROCfind((A%*2)+&400)
30160   NEXT
30170 END
30180 DEFPROCfind(D%)
30190 IF(!D%AND&FF00)=0THENENDPROC
30200 E%=!D%AND&FFFF
30210 I%=0
30220 REPEAT
30230   I%?&0900=I%?(E%+2)
30240   I%=I%+1
30250   UNTILI%?&08FF=0
30260 IFI%?E%=ASC"("THENPROCarray ELSEI%?&08FF=&D
30270 PRINTCHR$A%;$&0900;:IF I%?E%=ASC"(" THEN PRINT EL
SE PRINT"=";EVAL(CHR$A%+$&0900)
30280 PROCfind(E%)
30290 ENDPROC
30300 DEFPROCarray
30310 LOCALJ%,L%,K%,M%
30320 J%=I%+E%+2:L%=I%+&900:?L%=&D
30330 K%=?J%
30340 FORM%=1TOK%-1STEP2
30350   $&0900=$&0900+STR$(M%?J%-1)+","
30360   NEXT
30370 ?(&08FF+LEN$&0900)=ASC")"
30380 ENDPROC
/
```

BBC Micro owners will be used to designing characters – however, they might well have found great difficulty in getting a good program with which to do so. Electron owners will be new to designing characters and so could do with a little help.

Below we present our Character Designer program. It is capable of designing characters 134 to 159 and provides you with full editing facilities including inverse, row fill/un-fill, instant blanking and the ability to place any defined character in a grid. If you are using a BBC Micro with an Epson printer (or have a printer interface for your Electron with an Epson) you have the option of printing the characters out onto printer with a picture of each character as well as the VDU23 command needed.



Also the option of saving the new character set is given.

The program operates in MODE1 and depends on a series 1 OS to the extent of the character numbers. If you have the old 0.1 OS then changing the character numbers up should provide you with a working program. Otherwise, the program will run both on the BBC Micro and Acorn Electron.

To start off with, you are given a blank grid in the top left hand corner, a part of the screen set away in the right hand corner, the middle displaying characters 134–159 as they currently are and a question at the bottom asking which character you want to re-define. After choosing a valid character, the keys are:

CURSOR KEYS – Move cursor around the grid
TAB (BBC Micro) – Set or reset pixel
FUNC (Electron) – Set or reset pixel
SPACE – Clear grid
I – Inverse character
SHIFT–CURSOR LEFT – Fill row
SHIFT–CURSOR RIGHT – Unfill row
SHIFT–CURSOR UP – Fill column
SHIFT–CURSOR DOWN – Unfill column
RETURN – End edit

Once you have defined a character, you have the option of placing it in the box at the top right hand corner of the screen. If you choose to, the keys are:

CURSOR KEYS – Move cursor around the box
D – Delete character at cursor position
SPACE – Clear box
RETURN – End edit

You can safely move the cursor over a character in the box. This process will be repeated until you end the session by pressing ESCAPE. You then have the option to print out the characters to an Epson printer (you should delete lines 390–430 and 3060–3480 if you do not have an Epson printer/have an Electron). Finally, you can save the character set onto tape or disk (the filename used is "Chars". The characters set can be re-loaded into the computer by entering *LOAD Chars C00.

If you have a disk system, PAGE must be set to at least &1100 for the program to run.

```
These characters were designed using the LASERBUG Character Designer

To define character 134 as ↑ use VDU23,134,16,56,124,254,16,16,16,16
To define character 135 as ↗ use VDU23,135,31,15,7,11,17,32,64,0
To define character 136 as → use VDU23,136,8,12,14,255,14,12,8,0
To define character 137 as ↘ use VDU23,137,0,64,32,17,11,7,15,31
To define character 138 as ↓ use VDU23,138,8,8,8,8,127,62,28,8
To define character 139 as ↙ use VDU23,139,0,2,4,136,208,224,240,248
To define character 140 as ← use VDU23,140,0,16,48,112,255,112,48,16
To define character 141 as ↖ use VDU23,141,248,240,224,208,136,4,2,0
To define character 142 as ▓ use VDU23,142,255,255,255,255,255,255,255,255
To define character 143 as ▒ use VDU23,143,0,126,126,126,126,126,126,0
To define character 144 as ▪ use VDU23,144,0,0,60,60,60,60,0,0
To define character 145 as · use VDU23,145,0,0,0,24,24,0,0,0
To define character 146 as ● use VDU23,146,60,126,255,255,255,255,126,60
To define character 147 as ▒ use VDU23,147,0,60,126,126,126,126,60,0
To define character 148 as • use VDU23,148,0,0,24,60,60,24,0,0
To define character 149 as ↙ use VDU23,149,0,1,2,4,136,80,32,0
To define character 150 as ╳ use VDU23,150,129,66,36,24,24,36,66,129
To define character 151 as ⎍ use VDU23,151,96,210,76,66,68,68,68,0
To define character 152 as · use VDU23,152,0,0,0,24,24,0,0,0
To define character 153 as ·. use VDU23,153,0,96,96,0,0,6,6,0
To define character 154 as ·. use VDU23,154,192,192,0,24,24,0,3,3
To define character 155 as :: use VDU23,155,0,102,102,0,0,102,102,0
To define character 156 as :· use VDU23,156,195,195,0,24,24,0,195,195
To define character 157 as ## use VDU23,157,102,102,0,102,102,0,102,102
To define character 158 as ♣ use VDU23,158,0,15,31,31,127,204,204,255
To define character 159 as ▒ use VDU23,159,0,224,255,224,248,204,206,255
```

Continued from back page

POSTAGE RATES
A – Postage included in the price
B – 20p P & P
C – 40p P & P
D – 60p P & P
E – £1.00 P & P
F – £7.00
G – No postage on cables as long as they are ordered at the same time as a printer/monitor
H – Please include an A4 size SAE with at least 21p postage
I – £2.00 P & P
Overseas surface mail postage at twice U.K. prices (postage is taken into account with memberships). Overseas air mail goods (except memberships) by special arrangement only.

AIR MAIL POSTAL BANDS
Postal Band A includes Bahrain, Egypt, Israel, Oman, Saudi Arabia, United Arab Emirates
Postal Band B includes Canada, Falkland Islands, Hong Kong, India, Malaysia, South Africa, Sri Lanka, United States of America
Postal Band C includes Australia and New Zealand

\+ In the event of magazines being delayed, memberships to LASERBUG last until you have received the correct number of magazines (i.e. 12 or 6)
++ A list of the articles contained in these issues is available on request
+++ Printer types: A – Dot Matrix
B – Thermal
C – Ink Jet
D – 4 Colour Printer/Plotter
F – Daisywheel
G – 7 Colour Ink Jet
I – 7 Colour Dot Matrix
Warranty on printers varies between makes.
Full information is available on any individual printer(s). Please enquire at the usual address stating what printer(s) you are interested in.
++++ Monitor types: A – 14", Horizontal resolution 400 pixels
B – 14", Horizontal resolution 600 pixels
C – 14", Horizontal resolution 800 pixels
D – 14", 12 MHz Bandwidth
One year warranty on all monitors.
+++++ Requires DFS Upgrade (available from most Acorn dealers). Format utility disk supplied. Single sided disks, 100K. One year warranty. Full information is available on request – enquiries to the usual address.

TERMS
Cheques/POs should be made payable to LASERBUG. Educational orders accepted (not on Printers, Monitors and Disks), otherwise cash with order only. Overseas orders – Cheques/POs should be for £ Pounds Sterling, foreign currencies including Irish Phunts not accepted. No credit cards. VAT is included where appropriate.

```
/L.
   10 IFPAGE>&1100THENPRINT"Page must be moved down to
at least""&1100 - please re-load program":END
   20 ONERRORGOTO3580
   30 :
   40 REM CHARACTER DESIGNER
   50 REM  by Paul Barbour
   60 :
   70 REM    Version 1.1
   80 :
   90 REM For BBC Micro and
  100 REM  Acorn Electron
  110 :
  120 REM (c) LASERBUG 1984
  130 :
  140    :::::
  150 :
  160 MODE1
  170 VDU23,1,0;0;0;0;
  180 :
  190    :::::
  200 :
  210 PROCchar
  220 PROCgrid
  230 PROCbox
  240 PROCvar
  250 ONERRORGOTO370
  260 PROClist
  270 PROCstart
  280 PROCdecision
  290 REPEAT
  300   PROCmove
  310   PROCprint
  320   UNTILend%=TRUE
  330 PROCtidy
  340 PROCin_box
  350 end%=FALSE
  360 GOTO260
  370 ONERRORGOTO3580
  380 :
  390 REM Delete line 430 if
  400 REM you do not have a
  410 REM printer/have an Electron
  420 :
  430 PROCprintout
  440 PROCsave
  450 RUN
  460 :
  470    :::::
  480 :
  490 DEFPROCchar
  500 VDU23,128,255,129,129,129,129,129,129,255
  510 VDU23,129,255,129,189,189,189,189,129,255
  520 VDU23,130,0,0,0,0,0,0,0,255
  530 VDU23,131,128,128,128,128,128,128,128,128
  540 VDU23,132,255,0,0,0,0,0,0,0
  550 VDU23,133,1,1,1,1,1,1,1,1
  560 ENDPROC
  570 :
  580 DEFPROCgrid
  590 VDU19,0,6,0,0,0
  600 VDU19,2,4,0,0,0
  610 PRINTTAB(0,0);
  620 FORrow%=1TO8
  630   COLOUR1
  640   FORcolumn%=1TO8
  650     VDU128
  660     NEXT
  670   COLOUR2
  680   PRINT" 0  "
  690   NEXT
  700 ENDPROC
  710 :
  720 DEFPROCvar
  730 DIMchar%(7,7)
  740 DIMmem%(7)
  750 DIMbox%(25,8)
  760 FORfill%=0TO25
  770   FORfill2%=0TO8
  780     box%(fill%,fill2%)=32
  790     NEXT
  800   NEXT
  810 X%=0:Y%=0
  820 *FX4,1
  830 end%=FALSE
  840 ENDPROC
  850 :
  860 DEFPROCstart
  870 COLOUR2
  880 IFchar%(0,0)=0THENPRINTTAB(0,0);CHR$128:ELSEPRINT
TAB(0,0);CHR$129
  890 COLOUR1
  900 PRINTTAB(3,8);CHR$130
  910 PRINTTAB(4,9);CHR$131
  920 PRINTTAB(3,10);CHR$132
  930 PRINTTAB(2,9);CHR$133
  940 ENDPROC
  950 :
  960 DEFPROCmove
  970 REPEATmove%=INKEY(0):UNTIL(move%>135ANDmove%<140)
ORmove%=32ORmove%=73ORmove%=105ORmove%=13ORINKEY(-65)=-
1OR(move%>93ANDmove%<97)ORmove%=124ORmove%=9
  980 IFINKEY(-65)=-1THENmove%=9
  990 COLOUR1
 1000 IFchar%(X%,Y%)=0THENPRINTTAB(X%,Y%);CHR$128:ELSEP
RINTTAB(X%,Y%);CHR$129
 1010 IFmove%=13ORmove%=9THENchar%(X%,Y%)=char%(X%,Y%)
EOR1
 1020 IFmove%=32THENPROCclear
 1030 IF(move%=136ANDINKEY-1)ORmove%=94THENPROCcolumnon
 1040 IF(move%=137ANDINKEY-1)ORmove%=124THENPROCcolumno
ff
 1050 IF(move%=138ANDINKEY-1)ORmove%=95THENPROCrowoff
 1060 IF(move%=139ANDINKEY-1)ORmove%=96THENPROCrowon
 1070 IFmove%=73ORmove%=105THENPROCinverse
 1080 IFmove%=13THENend%=TRUE:move%=0
 1090 IFmove%=136THENX%=X%-1
 1100 IFmove%=137THENX%=X%+1
 1110 IFmove%=138THENY%=Y%+1
 1120 IFmove%=139THENY%=Y%-1
```

```
 1130 IFX%<0THENX%=0
 1140 IFX%>7THENX%=7
 1150 IFY%<0THENY%=0
 1160 IFY%>7THENY%=7
 1170 COLOUR2
 1180 IFchar%(X%,Y%)=0THENPRINTTAB(X%,Y%)CHR$128:ELSEPR
INTTAB(X%,Y%);CHR$129
 1190 COLOUR1
 1200 PROCbinary(Y%)
 1210 ENDPROC
 1220 :
 1230 DEFPROCbinary(row%)
 1240 COLOUR2
 1250 sum%=0
 1260 FORadd%=0TO7
 1270   IFchar%(add%,row%)=1THENsum%=sum%+(2^(7-add%))
 1280   NEXT
 1290 PRINTTAB(9,row%);sum%;"   "
 1300 mem%(row%)=sum%
 1310 ENDPROC
 1320 :
 1330 DEFPROCprint
 1340 VDU23,define%,mem%(0),mem%(1),mem%(2),mem%(3),mem
%(4),mem%(5),mem%(6),mem%(7)
 1350 PRINTTAB(3,9);CHR$define%
 1360 COLOUR1
 1370 PRINTTAB(5,9);"CHR$";define%
 1380 ENDPROC
 1390 :
 1400 DEFPROCclear
 1410 PRINTTAB(0,0);
 1420 PROCgrid
 1430 FORrow%=0TO7
 1440   FORcolumn%=0TO7
 1450     char%(row%,column%)=0
 1460     NEXT
 1470   NEXT
 1480 FORrow%=0TO7
 1490   mem%(row%)=0
 1500   NEXT
 1510 ENDPROC
 1520 :
 1530 DEFPROClist
 1540 FORlist%=134TO159STEP2
 1550   COLOUR1
 1560   PRINTTAB(0,(list%/2)-56)list%;" ";
 1570   COLOUR2
 1580   PRINTCHR$list%;TAB(20);
 1590   COLOUR1
 1600   PRINTlist%+1;" ";
 1610   COLOUR2
 1620   PRINTCHR$(list%+1)
 1630   NEXT
 1640 ENDPROC
 1650 :
 1660 DEFPROCcolumnon
 1670 COLOUR1
 1680 FORcolumn%=0TO7
 1690   char%(column%,Y%)=1
 1700   PRINTTAB(column%,Y%);CHR$129
 1710   NEXT
 1720 char%=0
 1730 ENDPROC
 1740 :
 1750 DEFPROCcolumnoff
 1760 COLOUR1
 1770 FORcolumn%=0TO7
 1780   char%(column%,Y%)=0
 1790   PRINTTAB(column%,Y%);CHR$128
 1800   NEXT
 1810 char%=0
 1820 ENDPROC
 1830 :
 1840 DEFPROCrowoff
 1850 FORrow%=0TO7
 1860   char%(X%,row%)=0
 1870   COLOUR1
 1880   PRINTTAB(X%,row%);CHR$128
 1890   PROCbinary(row%)
 1900   NEXT
 1910 char%=0
 1920 ENDPROC
 1930 :
 1940 DEFPROCrowon
 1950 FORrow%=0TO7
 1960   char%(X%,row%)=1
 1970   COLOUR1
 1980   PRINTTAB(X%,row%);CHR$129
 1990   PROCbinary(row%)
 2000   NEXT
 2010 char%=0
 2020 ENDPROC
 2030 :
 2040 DEFPROCinverse
 2050 FORrow%=0TO7
 2060   FORcolumn%=0TO7
 2070     IFchar%(column%,row%)=0THENchar%(column%,row%
)=1:PRINTTAB(column%,row%)CHR$129:ELSEchar%(column%,row
%)=0:PRINTTAB(column%,row%);CHR$128
 2080     NEXT
 2090   PROCbinary(row%)
 2100   COLOUR1
 2110   NEXT
 2120 char%=0
 2130 ENDPROC
 2140 :
 2150 DEFPROCdecision
 2160 COLOUR1
 2170 PRINTTAB(0,28)"What character (134-159) ?"
 2180 PRINT"> ";
 2190 COLOUR2
 2200 VDU23,1,1;0;0;0;
 2210 INPUT""define%
 2220 VDU23,1,0;0;0;0;
 2230 IFdefine%<134ORdefine%>159THENPRINTTAB(2,29);SPC(
80);TAB(2,29);:GOTO2200
 2240 PRINTTAB(0,28);SPC(100)
 2250 COLOUR1
 2260 start%=&C00+((define%-128)*8)
 2270 counter%=0
```

```
 2280 FORread%=start%TOstart%+7
 2290   data%=?read%
 2300   mem%(counter%)=data%
 2310   pos%=0
 2320   FORbreakdown%=7TO0STEP-1
 2330     binary%=2^breakdown%
 2340     IFdata%DIVbinary%=1THENchar%(pos%,(read%-star
t%))=1:PRINTTAB(pos%,(read%-start%));CHR$129:ELSEchar%(
pos%,(read%-start%))=0:PRINTTAB(pos%,(read%-start%));CH
R$128
 2350     data%=data%MODbinary%
 2360     pos%=pos%+1
 2370     NEXT
 2380   PROCbinary(counter%)
 2390   COLOUR1
 2400   counter%=counter%+1
 2410   NEXT
 2420 COLOUR2
 2430 PROCprint
 2440 PROCstart
 2450 ENDPROC
 2460 :
 2470 DEFPROCbox
 2480 COLOUR2
 2490 FORside%=0TO7
 2500   PRINTTAB(14,side%);CHR$133
 2510   NEXT
 2520 FORside%=15TO39
 2530   PRINTTAB(side%,8);CHR$132
 2540   NEXT
 2550 ENDPROC
 2560 :
 2570 DEFPROCtidy
 2580 COLOUR1
 2590 IFchar%(X%,Y%)=1THENPRINTTAB(X%,Y%);CHR$129:ELSEP
RINTTAB(X%,Y%);CHR$128
 2600 X%=0:Y%=0
 2610 ENDPROC
 2620 :
 2630 DEFPROCin_box
 2640 COLOUR1
 2650 PRINTTAB(0,28);"Do you want to place the characte
r in"
 2660 PRINT"box ? (Press ""Y"" or ""N"")"
 2670 REPEATbox$=GET$:UNTILINSTR("YyNn",box$)
 2680 PRINTTAB(0,28);SPC(80);
 2690 IFbox$="N"ORbox$="n"THENENDPROC
 2700 BX%=0:BY%=0
 2710 end%=FALSE
 2720 COLOUR2
 2730 PRINTTAB(BX%+15,BY%);CHR$define%
 2740 REPEAT
 2750   REPEATmove%=INKEY(0):UNTIL(move%>135ANDmove%<14
0)ORmove%=13ORmove%=32ORmove%=68ORmove%=100ORINKEY-65=-
10Rmove%=9
 2760   IFINKEY(-65)THENmove%=9
 2770   COLOUR1
 2780   PRINTTAB(BX%+15,BY%);CHR$box%(BX%,BY%)
 2790   IFmove%=9ORmove%=135THENbox%(BX%,BY%)=define%:m
ove%=0
 2800   IFmove%=13THENend%=TRUE:UNTILTRUE:ENDPROC
 2810 IFmove%=68ORmove%=100THENbox%(BX%,BY%)=32:move%=0
 2820 IFmove%=32THENPROCblank
 2830 IFmove%=136THENBX%=BX%-1
 2840 IFmove%=137THENBX%=BX%+1
 2850 IFmove%=138THENBY%=BY%+1
 2860 IFmove%=139THENBY%=BY%-1
 2870 IFBX%<0THENBX%=0
 2880 IFBX%>24THENBX%=24
 2890 IFBY%<0THENBY%=0
 2900 IFBY%>7THENBY%=7
 2910 COLOUR2
 2920 PRINTTAB(BX%+15,BY%);CHR$define%
 2930 UNTILend%=TRUE
 2940 ENDPROC
 2950 :
 2960 DEFPROCblank
 2970 FORfill%=0TO25
 2980   FORfill2%=0TO7
 2990     box%(fill%,fill2%)=32
 3000     PRINTTAB(fill%+15,fill2%);" "
 3010     NEXT
 3020   NEXT
 3030 move%=0
 3040 ENDPROC
 3050 :
 3060 DEFPROCprintout
 3070 VDU23,1,0;0;0;0;
 3080 PRINTTAB(0,28);"Do you want a printout of the cha
racters";
 3090 PRINT"(Press ""Y"" or ""N"")";SPC(15);
 3100 REPEATbox$=GET$:UNTILINSTR("YyNn",box$)
 3110 IFbox$="N"ORbox$="n"THENENDPROC
 3120 VDU28,0,29,39,28
 3130 char%=134
 3140 VDU2
 3150 VDU1,15
 3160 PRINT"These characters were designed using the LA
SERBUG Character Designer"'
 3170 FORdump%=&C30TO&CFFSTEP8
 3180   PRINT"To define character ";char%;" as ";
 3190   counter%=0
 3200   print$="VDU23,"+STR$char%
 3210   FORread%=dump%TOdump%+7
 3220     data%=?read%
 3230     print$=print$+","+STR$data%
 3240     pos%=0
 3250     FORbreakdown%=7TO0STEP-1
 3260       binary%=2^breakdown%
 3270       IFdata%DIVbinary%=1THENchar%(pos%,(read%-du
mp%))=1:ELSEchar%(pos%,(read%-dump%))=0
 3280       data%=data%MODbinary%
 3290       pos%=pos%+1
 3300       NEXT
 3310     counter%=counter%+1
 3320     NEXT
 3330   VDU1,27,1,75,1,8,1,12?
 3340   FORX%=0TO7
 3350     mem%(X%)=0
 3360     FORY%=0TO7
```

```
 3370       IFchar%(X%,Y%)=1THENmem%(X%)=mem%(X%)+(2^(7
-Y%))
 3380       NEXT
 3390     VDU1,mem%(X%)
 3400     NEXT
 3410   PRINT" use ";print$
 3420   char%=char%+1
 3430   NEXT
 3440 VDU3
 3450 CLS
 3460 VDU26
 3470 ENDPROC
 3480 :
 3490 DEFPROCsave
 3500 PRINTTAB(0,28);"Save character set onto tape/disk
?     ";
 3510 PRINT"(Press ""Y"" or ""N"")"
 3520 REPEATsave$=GET$:UNTILINSTR("YyNn",save$)
 3530 IFsave$="N"ORsave$="n"THENENDPROC
 3540 VDU28,0,31,39,28
 3550 *SAVE Chars C00+FF
 3560 VDU26
 3570 ENDPROC
 3580 MODE7
 3590 PRINTCHR$131;"Error:"
 3600 REPORT
 3610 PRINT'"(Error number ";ERR;")"
 3620 PRINT"At line ";ERL
 3630 *FX4
 3640 PRINT'CHR$129;"Program stopped"
 3650 PRINT
```

simple sprites

What is a sprite? Well, a sprite is defined as "a user defined character which can be moved around the screen by the use of comparatively simple instructions. These instructions obviate the need to draw the shape, undraw it, shift, draw again and so on". A sprite is also normally multi-coloured and provides a fairly easy way to produce good games, etc. Several home computers have the ability to produce sprites, unfortunately the BBC Micro is not one of them.

Although in the space I have here I cannot produce a program to produce full, multi-coloured sprites, we have tried our best to produce something approaching the whole concept of sprites. To start off with you will need to type in the following program which consists of two procedures and a function (remember that functions return a value, procedures do not). You can of course leave out lines 9910–9990 (as long as you don't forget where the program came from!). Briefly, PROCinitialise sets up the arrays needed for the sprites, PROCsetup sets up each sprite and defines its characteristics and FNmove is the actual section that moves them.

```
>L.
 9910 REM SIMPLE SPRITES
 9920 REM by Paul Barbour
 9930 :
 9940 REM Version 1.0
 9945 :
 9950 REM BBC Micro & Electron
 9955 :
 9960 REM (c) LASERBUG 1984
 9970 :
 9980    ::::::
 9990 :
10000 DEFPROCinitialise(q%)
10010 DIMc%(q%),x(q%),y(q%),a(q%),b(q%),g%(q%),f%(q%)
10020 ENDPROC
10030 :
10040 DEFPROCsetup(q%,c%,x,y,q%,r%,v%)
10050 c%(q%)=c%
10060 x(q%)=x
10070 y(q%)=y
10080 a(q%)=SINRAD(r%)*v%
10090 b(q%)=COSRAD(r%)*v%
10100 q%(q%)=q%
10110 f%(q%)=0
10120 VDU5
10130 GCOL3,g%(q%)
10140 MOVEx(q%),y(q%)
10150 VDUc%(q%)
10160 ENDPROC
10170 :
10180 DEFFNmove(q%)
10190 IFf%(q%)=2THEN10330
10200 VDU5
10210 GCOL3,g%(q%)
10220 MOVEx(q%),y(q%)
10230 *FX19
10240 VDUc%(q%)
10250 IFf%(q%)=1THENVDU4:f%(q%)=2:GOTO10330
10260 x(q%)=x(q%)+a(q%)
10270 y(q%)=y(q%)+b(q%)
10280 GCOL3,g%(q%)
10290 MOVEx(q%),y(q%)
10300 *FX19
10310 VDUc%(q%)
10320 IF(x(q%)<0ORx(q%)>1280)OR(y(q%)<0ORy(q%)>1024)THE
Nf%(q%)=1
10330 =f%(q%)
```

The easiest way to explain sprites is to have an example running and so try the following example (the best thing to do probably is type in the above, save it and add on the following programs to it).

```
>L.
  10 REM Sprite Demo 1
  20 :
  30 MODE4
  40 VDU23,1,0;0;0;0;
  50 VDU23,128,0,60,126,126,126,126,60,0
  60 PROCinitialise(1)
  70 PROCsetup(1,128,640,512,1,RND(360),16)
  80 REPEAT
  90   status%=FNmove(1)
 100   UNTILstatus%=2
 110 GOTO70
```

Firstly, of course, you need to set up the screen mode. Next, the actual sprite characters should be defined in the normal manner, i.e. with a VDU23 command. Any character numbers can of course be used (within the limits of the normal BBC Micro OS of course). PROCinitialise defines how many sprites you are going to use. Because this defines arrays (arrays cannot be re-defined) this PROC can only be used once in the program (unless you use the CLEAR command which wipes all variables) so do make sure you get this number right from the outset. If at any time you get a Subscript error, this is most likely because you did not set up enough sprites (i.e. you are using a number outside the sprite/array range). PROCsetup needs to be executed once for each character. In order it requires to know (i) which sprite you are setting up, (ii) which character you want to use as the sprite, (iii) the X and Y co-ordinates of the start position to graphics resolution, i.e. 1280×1024, (iv) the colour of the sprite (although the sprite will be EORed on screen and so the colour will be changed if it passes over a coloured object), (v) the direction the sprite is to travel (0–360') and the speed of the sprite (try 16 as a first go and alter from there). The speed of the sprite will of course be slowed down by the number of objects that are moving. PROCsetup can be used as many times as needed in the program. To actually move the sprite you use FNmove, including the sprite number of course. FNmove cannot be used alone as a PROCedure and must be assigned some value, i.e. status=FNmove(1). FNmove returns a value depending on the sprites status – 0 when it is moving, 1 when it has reached the edge of the screen and 2 when it has finished moving. The sprite should be restarted when status=2. If you require more detailed information on what the sprite is doing then the arrays used are:

c% – Character number
x% – X position of sprite
y% – Y position of sprite
a% – X movement of sprite
b% – Y movement of sprite
g% – Colour of sprite
f% – Value returned by function

Hopefully the simple sprite program is quite transparent and should be open to alteration. As a final demonstration, try the following program:

```
10 REM Sprite Demo 2
20 :
30 MODE2
40 VDU23,1,0;0;0;0;
50 :
60 VDU23,128,0,56,127,248,255,85,42,0
70 VDU23,129,24,24,24,24,60,102,24,0
80 VDU23,130,24,60,60,255,165,255,126,0
90 VDU23,131,60,126,254,254,254,254,126,60
100 :
110 PROCstars
120 PROCmountains
130 PROCinitialise(5)
140 PROCsetup(1,128,0,32,7,90,8)
150 PROCsetup(2,129,RND(1280),0,2,0,32)
160 PROCsetup(3,130,0,RND(512)+512,3,90,16)
170 PROCsetup(4,131,1280,RND(512)+512,5,225+RND(90),6
4)
180 REPEAT
190   status1%=FNmove(1)
200   status2%=FNmove(2)
210   status3%=FNmove(3)
220   status4%=FNmove(4)
230   IFstatus1%=1THENPROCsetup(1,128,0,32,7,90,8)
240   IFstatus2%=2THENPROCsetup(2,129,RND(1280),0,2,0
,32)
250   IFstatus3%=2THENPROCsetup(3,130,0,RND(512)+512,
3,90,16)
260   IFstatus4%=2THENPROCsetup(4,131,1280,RND(512)+5
12,5,225+RND(90),64)
270   UNTIL0
280 :
290 DEFPROCstars
300 FORX%=0TO250
310   GCOL0,RND(8)+7
320   PLOT69,RND(1280),RND(1024)
330   NEXT
340 ENDPROC
350 :
360 DEFPROCmountains
370 GCOL0,1
380 top%=RND(256)+256
390 bottom%=RND(256)
400 FORX%=0TO1280STEP128
410   MOVEX%,0:DRAWX%,bottom%
420   PLOT85,X%+64,top%
430   DRAWX%+64,0
440   PLOT85,X%,0
450   bottom%=RND(256)
460   MOVEX%+64,0:DRAWX%+64,top%
470   PLOT85,X%+128,bottom%
480   DRAWX%+128,0
490   PLOT85,X%+64,0
500   top%=RND(256)+256
510   NEXT
520 ENDPROC
```

r h light pen

There are a few light pens around for the BBC Micro – the reason of course being that they are so easy to interface to the Beeb via the analogue port. The nearest there is to an official or accepted pen is the one made by R.H. Electronics.

The pen is not a "standard" light pen as such – neither is it merely a simple light pen that plugs straight into the analogue port. The pen itself consists of a small circuit board 0.5″ × 4″ with various components on. The pen in turn is connected to an interface box with a much larger board (5.5″ × 4.4″) with various components including 11 chips! All the circuit boards are very neatly finished. At least you get something substantial for your money.

Externally the interface box measures 4.5″ × 6″ × 1.5″ and sits quite neatly on the right of your computer – it is coloured cream like the BBC Micro and blends in very well. A 15″ long lead connects the box to the analogue port leaving the pen on the end of a lead just over 3′.

The pen itself is quite comfortable for an adult to hold. The nose of the pen forms a switch – you register the pen to the computer by touching it on the screen. On the back is a small LED which lights up when valid data is at the end of the pen.

To use the pen in your own programs, you need to use a special machine code driver and a few lines of BASIC. Before we move onto that though, first we will look at the software that comes with the pen.

All programs come in two parts – the main program and the machine code driver. The first one SETUP allows you to calibrate the pen. First of all a frame position test allows you to check what the correct value of *TV should be. The next is sensitivity which checks that the pen can pick up data – the sensitivity is preset and should not need altering but, if necessary, then this can be done by adjusting a small potentiometer – in older versions this is inside the case but on the latest pens this is accessible externally. Finally, there is trim check which will see if a vertical adjustment is necessary. It is up to you to adjust *TV and the sensitivity is adjusted in hardware, but some programs that use the light pen ask for the trim factor to be entered.

The next program is a simple drawing program which will allow you to draw lines over the screen. The program is very crude but just an example of what you can do.

Noughts and crosses is the third item which of course uses the light pen to input the moves – the game is for two people and you will find if you are not too careful that you press the pen twice by mistake and make the other person forfeit his go.

The final "program" as such is a simple utility allowing you to program the various attributes of the pen. The last item on the tape is the machine code driver itself which allows you to write the pen into your own programs.

All of these programs are really very simple and are just examples of what can be done. There is (or was at the time of writing) no software available for the pen, other than the piece supplied, although high quality graphics software is promised, including a technical drawing and freehand drawing program. So, the main idea of buying the pen is to use it yourself in your own programs. How easy is it to use?

Well, I won't quite say it's like reading through fog, but certainly the manual is like reading through a heavy mist. It is certainly not written in an informative or instructive way – the manual does give you all the information, but certainly not in a way friendly to the newcomer.

The machine code driver can be located anywhere in the memory (within the normal confines of the computer of course – say &2C00. Then a number of variables can be set up based on this location and it is from here that everything is based – there are 28 variables that can be set up in all, although not all are necessary. Although it is not as simple as this, the X position of the light pen can be read with ?P_xtext and the Y position with ?P_ypoint. You can read whether the switch at the end of the pen is pressed by looking at ?P_switch. In a future issue we will be looking at exactly how to use the R.H. Electronics light pen, including a graphics program.

In summary, the pen is quite robust and works well. The manual unfortunately is appalling, but hopefully our forthcoming article should help somewhat. No backup software is available yet but hopefully this should change by the New Year. Light pens seem to give the impression to people that you plug them in and then you can paint pictures on the screen. A light pen can do this – but only with the right program behind it. As it stands, it is most suitable for incorporating in your own programs, but a confused manual means you will have to do quite a bit of work to get into it. The programs included on the tape are not protected and so you can disect them to get more help.

Hopefully, when the backup becomes available, this pen should prove a good buy.

easy speech

The speech command on the BBC Micro takes the format SOUND a,b,c,d where a is always –1, c and d are always 0 and only b is of any real significance. So, to say for instance ACORN COMPUTERS requires the line SOUND–1,160,0,0:SOUND–1,179,0,0:SOUND–1,138,0,0 when the real "bones" of the matter are 160, 179 and 138. With this in mind, below is a listing of PROCtalk along with PROCsettalk, which makes it easier to use.

PROCsettalk simply sets up two function keys – key 2 to produce PROCtalk(" and key 6 @");M. PROCtalk requires a list of word numbers – after each number you should have a "–". So, if we wanted to say ACORN COMPUTERS then you would press f2 to get PROCtalk(", and then enter 160–, 179– and 138–. Finally, to end you press f6 to get @") <return>. In short, this gives:

```
PROCtalk("160-179-138-@")
```

which would say words 160, 179 and 138 thus giving Acorn Computers.

This method might seem a little complex, but once you get used to it, you will find it a much easier way to enter speech.

In a month or two we will have a full speech editor. . .

```
>L.
  10 REM Easy Speech
  20 REM by Paul Barbour
  30 :
  40 REM For BBC Micro with
  50 REM Speech Upgrade
  60 :
  70 REM Version 1.0
  80 :
  90 REM (c) LASERBUG 1984
 100 :
 110    :::::
 120 :
 130 PROCsettalk
 140 PROCtalk("270-247-209-279-127-202-@")
 150 END
 160 :
 170 DEFPROCsettalk
 180 *KEY2PROCtalk("
 190 *KEY6@")|M
 200 ENDPROC
 210 :
 220 DEFPROCtalk(speech$)
 230 LOCALpointer%,speak$,speak%
 240 pointer%=1
 250 speak$=""
 260 REPEAT
 270   speak$=speak$+MID$(speech$,pointer%,1)
 280   pointer%=pointer%+1
 290   UNTILRIGHT$(speak$,1)="-"ORRIGHT$(speak$,1)="@"
 300 IFRIGHT$(speak$,1)="@"THENENDPROC
 310 speak%=VAL(LEFT$(speak$,LEN(speak$)-1))
 320 IFspeak%<32ORspeak%>291THENPRINT"Speech parameter
too large":END
 330 SOUND-1,speak%,0,0
 340 GOTO250
>
```

IN
CONJUNCTION
WITH

PRESENTS

SOFTWARE SEARCH

Have you written a REALLY good program lately? If so what are you going to do with it. Nothing? Send it to a magazine for maybe £10 or £20? Attempt to sell it yourself with all the risks that involves? Enter it in the LASERBUG Software Search competition and possibly win an RGB Monitor? If you've any sense then there is only one answer!

Quite simply, LASERBUG is searching for some good software – the kind that you don't run once and then discard but keep using over and over again. We would like to see all kinds of programs - games educational, business and utilities.

The programs will be judged on their individual merits and the winner will receive an RGB Monitor donated by Cabel Electronics. There will be one monitor to win each month.

Programs may be sent either on cassette or disk. If on cassette, please supply a copy of both 300 and 1200 baud. For disks, we will accept both 40 and 80 tracks on either single or double sided disks. Please do not send us your only copy of the program as we will not be able to return any. Make sure your name and address is on the cassette/disk and any accompanying documentation. Employees/relations of employees of either LASERBUG or Cabel Electronics are not eligible for entry in this competition, as are non-members of LASERBUG. There is no cash alternative for the prize. The closing date for Software 4 is the last working day in March. The winner will be notified by post and their program may be printed in LASERBUG, included in a Software Library or perhaps both. In both events the author will be acknowledged but no further payment made. The program must be the authors own, unaided work and should not have been submitted elsewhere. Judging will be carried out by the LASERBUG editor and the editor's decision is final. No correspondence will be entered into with regards these rules.

SOFTWARE SEARCH 4 ENTRY FORM

I enclose a program for entry to the LASERBUG Software Search competition, the details of which are:

PROGRAM NAME ..... REQUIREMENTS.....

PROGRAM TYPE
- ☐ GAME
- ☐ EDUCATION
- ☐ BUSINESS
- ☐ UTILITY
- ☐ OTHER *(Please specify)*

_____

MEDIA
- ☐ CASSETTE
- ☐ 40 TRACK DISK
- ☐ 80 TRACK DISK

PROGRAMMERS NAME ..... MEMBERSHIP NUMBER .....

ADDRESS.....

.....

I am a member of LASERBUG. The program I have submitted to this competition is my own, unaided work and has not been sent to any other organisation. I understand that if I win this compeition I will receive an RGB Monitor in exchange for full rights to the program. My program in turn may be printed in LASERBUG or included as part of a Software Library. In either case I will be acknowledged but not receive any further payment. I accept the rules laid down in this competition and agree that the editor's decision in judging the winner is final.

SIGNED..... DATE.....

☐ If I do not win Software Search 4, I would like my program to be entered for Software Search 5, 6, etc.

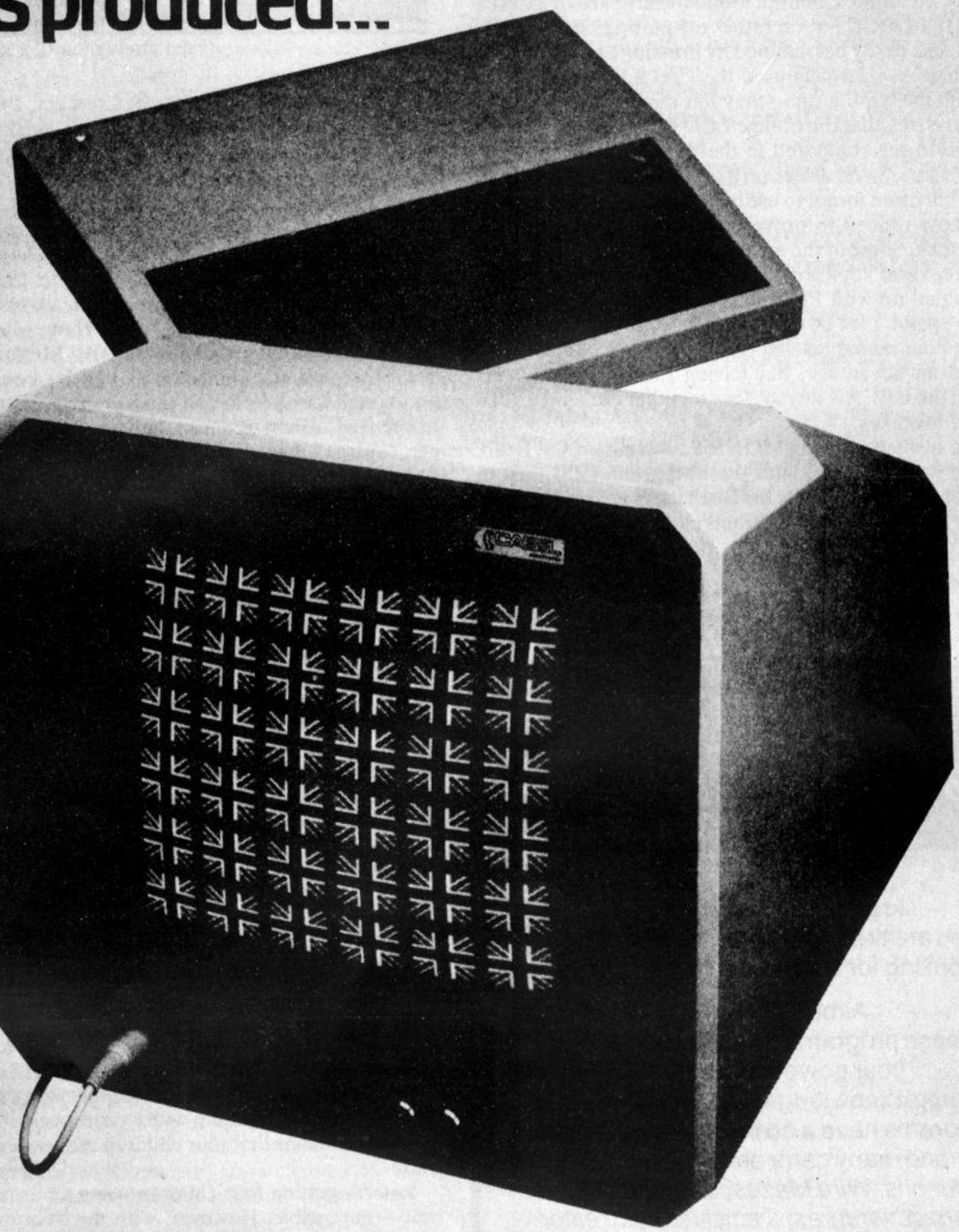
☐ If I do not win Software Search 4, I wish my program to be withdrawn from the competition.

THE FINAL CLOSING DATE FOR SOFTWARE SEARCH 4 IS THE LAST WORKING DAY IN MARCH

You may enter more than one program in the competition but each entry must be accompanied by a copy of this form. Your entry is not valid unless is signed and your membership number included.

Send completed form and program to Software Search, LASERBUG, 10 Dawley Ride, Colnbrook, Slough, Berks., SL3 0QH.

Style and sophistication combined with modern technology has produced...



A 14" British colour monitor at a price you really can afford. £199.50 plus VAT.

CABEL electronic

19 High Street, Tewkesbury, Gloucestershire GL20 5AW
Telephone: 0684 298840 Telex: 339671 ALD FAB

18 introducing the electron

One of the main markets for the Electron are school children who use the BBC Computer at school and so want to use the Acorn Electron at home. Also with the price of the Electron so low compared with the BBC Computer and the relatively similarity, it is also possible that schools will start filling their classrooms with Electrons as well as BBCs. How do children that have been using the BBC Micro adapt to the Electron?

The differences in hardware and software have been covered in LASERBUG elsewhere – we will look at the differences in operation.

The keyboard is obviously different and this provides the first stumbling block we found. Children found the absence of a CAPS LOCK and SHIFT LOCK button rather off-putting. Small letters on the Electron can easily be obtained by pressing SHIFT and the letter keys. When they actually used the CAPS LOCK function (using SHIFT-FUNC) quite often they felt they had to lean over the computer to check that the orange LED had actually gone out – it is very hard to see compared to the BBC Micro.

Everyone who used the Electron tended to like the special keyword entry but often forgot to use it. Having that many BASIC words on the keys tended to prove a memory jogger when the children were stuck. Some of the more ham-fisted children tended to end up with several keywords at once, i.e. when trying to get PRINT they ended up with PRINTPRINTPRINT which meant they tended to spend a lot of time on the delete key.

Editing got rather mixed results. Having the DELETE key on its own proved an advantage, but having the COPY key, the cursor keys and the BREAK key so close together did tend to lead to a number of mistakes.

In operation, instead of using MODE 7 as usual on the Beeb, they tended to use MODE 6 with the background changed to BLUE. This was soon followed by the Beeb users proving perhaps that the lines in MODE 6 make programs easier to follow as far as the children are concerned.

All in all, everyone who used the Electron after being used to the BBC Micro tended to soon adapt, taking advantage of the single keyword facility and in the end, a good number actually preferred using thhe Electron, although if that would change "when the novelty wears off" remains to be seen.

microwriting

A Microwriter is a special 6 key keyboard and, with those 6 keys, the full ASCII character set can be produced. Normally the Microwriter costs several hundred pounds, has its own built in LED screen, battery backed memory, RS232 interface and built in wordprocessing software. It is normally quite portable, meaning you can write your letters, etc. wherever you are, and when you get back to the office you plug in the printer and get a hard copy. The reason that people use them are because it is so easy to learn how to touch type on them. Now, Microwriter have produced a cut down version for the BBC Micro. The first keyboard costs £40 and additional onces can be bought for £30 – why additional ones? Well, you can have *four* Microwriters connected to the BBC Micro at once – and not just connected but each one can control the computer.

The Microwriter is designed to be sat on a lap or perhaps on the table. Your hand then fits quite naturally just above the top five keys, leaving the sixth key just below your thumb – this sixth key is used for additional functions, i.e. for editing, etc. . .

The principle of using a Microwriter is based on handwriting. You imagine the letter in the head that you are going to write and press the keys on the Microwriter that fit that shape. Say for instance that you are going to print the letter T. The main part of the letter H is the bar in the middle and so you make a vertical line with your fingers. The letter L has a bar at the bottom and a left hand side and so you press three keys – one where each corner of the letter would be and so on. This might sound complicated but it is in actual fact much easier to learn than say a normal QWERTY keyboard – where is the logic in the Beeb's keyboard. Why press a key in the top left hand corner for Q? Why not make the shape of the letter Q?

In reality, although it sounds as though you will take a long while to adapt, within an hour most people will be typing without the aid of prompt cards and within a few hours you should be touch typing – on the Microwriter. Microwriter themselves reckon that a 6 year old of average intelligence should be touch typing using a Microwriter within only half an hour.

A machine code driver program comes with the Microwriter. It means that any Microwriter keyboard can replace (or in fact be used with) the BBC keyboard. And if four keyboards are connected, four people can type a program at once. Not only can it be used to type programs and with wordprocessors such as Wordwise or View, it means that four children can use a program simultaneously.

You try getting four children using a normal keyboard at one time – impossible. However, with the Microwriter four children can share one computer, and more than sharing, by taking the computer in turns they can all answer it at once!

Touch typing is a skill which takes a long while to develop – mainly because the QWERTY keyboard isn't logical as such. Touch typing on the Microwriter is easy and can be learned quickly. It means that you can type in programs or use a wordprocessor much faster.

The only disadvantage with the Microwriter is that, unfortunately, you cannot use disks with the current software. If Microwriter updated their software to alter this then you would have a product that every BBC Micro owner could not be without. As it stands, it is an excellent product for educational uses. RECOMMENDED

BRAINTEASERS
FOR THE
BBC & ELECTRON
COMPUTERS

PROGRAMS
to puzzle and amuse
If you are tired of zap and pow games
and are looking for something different, **this is it**

Aimed at 15 +
these programs **demand** use of
your powers of logic,
general knowledge and deduction.
All programs have **a competition element**
and many carry an IQ rating.
Whodunnits; Wire Mazes; Safe Crackers;
Western adventures; Decisiveness Tester

Available from bookshops,
Acorn dealers only £5.95
or direct at £5.95 + 55p p/p from:
PHOENIX PUBLISHING ASSOCIATES, 14 VERNON ROAD, BUSHEY, HERTS, WD2 2JL. Tel WATFORD 0923 32109

NAME .....
ADDRESS .....
.....
..... **CODE** .....

icbm attack!!!

In this game you must defend your country from a wave of ICBMs rapidly heading in your direction. As the program stands it is ready for the BBC Micro with OS 1.2. If you are a Beeb owner with the 0.1 OS then you will (a) have to change the character definitions (i.e. 128 becomes 224, 129 becomes 225, etc.), (b) change the cursor off lines (190 and 1200) and (c) get rid of the *FX19 call at line 1520.

```
>L.  10 REM ICBM ATTACK !!!
     20 REM by Paul Barbour
     30 :
     50 REM Version 1.0
     60 :
     61 REM BBC Micro Only
     62 :
     70 REM (c) LASERBUG 1984
     80 :
     90    :::::
    100 :
    105 hit%=FALSE
    110 MODE1
    120 VDU23,1,0;0;0;0;
    130 :
    140    :::::
    150 :
    160 PROCset_up1
    170 PROCinstructions
    180 PROCset_up2
    190 :
    200 REPEAT
    210   IFRND(50)=1THENPROCscore
    220   IFfire%=TRUE THENPROCmove_gtam1
    230   PROCmove_base
    240   PROCmove_icbm
    250   IFfire%=TRUE PROChit
    260   IFfire%=TRUE THENPROCmove_gtam2
    270   IFfire%=FALSE THENPROCfire
    280   IFhit%=TRUE THENPROChit2
    290   UNTILmisses%=3
    300 PROCthe_end
    310 GOTO1990
    320 :
    330 DEFPROCset_up1
    340 VDU23,128,24,0,24,0,24,0,90,0
    350 VDU23,129,0,24,0,24,36,126,165,255
    360 VDU23,130,24,24,24,24,24,60,66,66
    370 VDU23,131,0,0,0,0,0,0,24,36
    380 VDU23,132,195,102,102,60,60,0,0,0
    390 VDU23,133,60,24,24,0,0,60,24,24
    400 VDU23,134,-1,-1,-1,-1,-1,-1,-1,-1
    410 VDU19,3,6,0;0;
    420 ENDPROC
    430 :
    440 DEFPROCset_up2
    450 fire%=FALSE
    460 basex%=20
    470 icbmx%=RND(38)
    480 icbmy%=1
    490 hits%=0:misses%=0
    500 reset%=FALSE:reset2%=FALSE
    510 PROCbase(basex%,30)
    520 PROCicbm(icbmx%,icbmy%)
    530 COLOUR2:PRINTTAB(0,0);"Missiles Destroyed: ";:COL
OUR3:PRINT;hits%:COLOUR2:PRINTTAB(30,0);"Hits: ";:COLOU
R3:PRINT;misses%
    540 ENDPROC
    550 :
    560 DEFPROCmove_base
    570 PRINTTAB(basex%,30);" "
    580 IFINKEY-98basex%=basex%-1
    590 IFINKEY-67basex%=basex%+1
    600 IFbasex%<0THENbasex%=0
    610 IFbasex%>38THENbasex%=38
    620 PROCbase(basex%,30)
    630 ENDPROC
    640 :
    650 DEFPROCmove_icbm
    660 PRINTTAB(icbmx%,icbmy%);" "
    670 icbmy%=icbmy%+1
    680 IFreset%=TRUE THENicbmy%=1:icbmx%=RND(38):reset%=
FALSE
    690 IFicbmy%=30 THENPROCbase_hit:misses%=misses%+1:CO
LOUR3:PRINTTAB(36,0);misses%:icbmy%=0:icbmx%=RND(38)
    700 PROCicbm(icbmx%,icbmy%)
    710 IFfire%=TRUE PROChit
    720 SOUND&11,-10,255-(icbmy%*8.5),10
    730 ENDPROC
    740 :
    750 DEFPROCfire
    760 IFINKEY(-1)THENfire%=TRUE:gtamy%=29:PROCgtam(base
(%,gtamy%)
    770 ENDPROC
    780 :
    790 DEFPROCmove_gtam1
    800 PRINTTAB(basex%,gtamy%);" "
    810 IFINKEY(-1)THENCOLOUR1:PRINTTAB(basex%,gtamy%);CH
R$134:SOUND&10,-15,5,2:COLOUR0:PRINTTAB(basex%,gtamy%);
" ":gtamy%=29
    820 ENDPROC
    830 :
    840 DEFPROCmove_gtam2
    850 gtamy%=gtamy%-1
    860 IFgtamy%<0ORreset2%=TRUE THENfire%=FALSE:reset2%=
FALSE:ENDPROC
    870 PROCgtam(basex%,gtamy%)
    880 SOUND&10,-7,4,3
    890 IFfire%=TRUE PROChit
    900 ENDPROC
    910 :
    920 DEFPROChit
    930 IFbasex%=icbmx% AND gtamy%=icbmy% THENhit%=TRUE
    940 ENDPROC
    950 :
    960 DEFPROChit2
    970 COLOUR1:PRINTTAB(icbmx%,icbmy%);CHR$134:COLOUR0:P
RINTTAB(icbmx%,icbmy%-1);" ":SOUND0,-15,6,5:PROCpause(1
0):hits%=hits%+1:COLOUR3:PRINTTAB(20,0);hits%:reset%=TR
UE:reset2%=TRUE:PRINTTAB(icbmx%,icbmy%);" ":hit%=FALSE
    980 ENDPROC
    990 :
   1000 DEFPROCscore
   1010 COLOUR2:PRINTTAB(0,0);"Missiles Destroyed: ";:COL
OUR3:PRINT;hits%:COLOUR2:PRINTTAB(30,0);"Hits: ";:COLOU
R3:PRINT;misses%
   1020 ENDPROC
   1030 :
```

```
1040 DEFPROCbase_hit
1050 IFicbmx%=basex%THENPROCbang:GOTO1990
1060 ENDPROC
1070 :
1080 DEFPROCover
1090 VDU5
1100 x%=4*(X%*8)
1110 y%=4*(255-(Y%*8))
1120 MOVEx%,y%:VDUchr%
1130 VDU4,23,1,0;0;0;
1140 ENDPROC
1150 :
1160 DEFPROCbase(X%,Y%)
1170 COLOUR3
1180 PRINTTAB(X%,Y%);CHR$128
1190 GCOL0,1
1200 chr%=129:PROCover
1210 ENDPROC
1220 :
1230 DEFPROCgtam(X%,Y%)
1240 COLOUR2
1250 PRINTTAB(X%,Y%);CHR$130
1260 GCOL0,1
1270 chr%=131:PROCover
1280 ENDPROC
1290 :
1300 DEFPROCicbm(X%,Y%)
1310 COLOUR1
1320 PRINTTAB(X%,Y%);CHR$132
1330 GCOL0,3
1340 chr%=133:PROCover
1350 ENDPROC
1360 :
1370 DEFPROCpause(sec%)
1380 TIME=0
1390 REPEATUNTILTIME=sec%
1400 ENDPROC
1410 :
1420 DEFPROCthe_end
1430 FORX%=0TO255
1440   SOUND&11,0,X%,0:SOUND&10,-15,7,1
1450   *FX19
1460   VDU19,RND(4),RND(4),0;0;
1470   NEXT
1480 FORZ%=1TO10
1490   FORX%=0TO3:VDU19,X%,7,0;0;:NEXT
1500   SOUND0,-15,6,3
1510   PROCpause(3)
1520   FORX%=0TO3:VDU19,X%,0,0;0;:NEXT
1530   SOUND0,-15,6,3
1540   PROCpause(3)
1550   NEXT
1560 ENDPROC
1570 :
1580 DEFPROCbang
1590 X%=4*(basex%*8)
1600 VDU29,X%;0;
1610 FORX=-15TO0STEP.2
1620   GCOL0,RND(3)
1630   MOVE0,0
1640   Y%=RND(180)-90
1650   Z%=RND(200)
1660   DRAWSINRADY%*Z%,COSRADY%*Z%
1670   SOUND0,X,RND(3)+3,1
1680   NEXT
1690 ENDPROC
1700 :
1710 DEFPROCinstructions
1720 CLS
1730 COLOUR3
1740 PRINTSTRING$(40,"=")
1750 COLOUR1
1760 PRINTTAB(12)"ICBM ATTACK !!!"
1770 COLOUR3
1780 PRINT'STRING$(40,"=")
1790 COLOUR2
1800 PRINT''"Help !!!  There is a whole wave of Inter"
1810 PRINT" Continental Balistic Missiles (ICBM's -"
1820 PRINT"   ) heading towards your missile base"
1830 PROCicbm(2,12):COLOUR2
1840 PRINT'" ( ).  Using your Ground To Air Missiles"
1850 PROCbase(2,14):COLOUR2
1860 PRINT'" (GTAM -  ) you must destroy as many as"
1870 PROCgtam(9,16):COLOUR2
1880 PRINT'"possible.  Just three hits on the ground"
1890 PRINT"or one hit on your base means the end..."
1900 PRINT''TAB(6)"Z <   > X              SHIFT"
1910 PROCbase(10,24)
1920 PROCgtam(26,24)
1930 COLOUR3:PRINT'''TAB(10)"Press ";:COLOUR1:PRINT"SP
ACE ";:COLOUR3:PRINT"To Start"
1940 REPEATUNTILINKEY(-99)
1950 CLS
1960 ENDPROC
1970 :
1980 REM Final Score
1990 CLS:VDU20
2000 A$="FINAL SCORE: "+STR$(hits%)
2010 FORZ%=1TOLEN(A$)
2020   N%=ASC(MID$(A$,Z%,1))
2030   A%=10:X%=0:Y%=10
2040   C%=&A00
2050   ?C%=N%
2060   CALL&FFF1
2070   VDU23,135,C%?1,C%?1,C%?2,C%?2,C%?3,C%?3,C%?4,C%
?4
2080   VDU23,136,C%?5,C%?5,C%?6,C%?6,C%?7,C%?7,C%?8,C%
?8
2090   VDU135,10,8,136,11
2100   NEXT
2110 PROCpause(500)
2120 RUN
```

Supposing you want to sell a program for the BBC Micro. The obvious thing to do is instead of just saying it works on the BBC Micro, say it works on the BBC Micro AND Acorn Electron. Is this wise? Well, the answer is no. Although the Electron is largely compatible with the BBC Micro, there are enough differences between the two to say that you must check your program out on the Electron. If you write the program on the Beeb and then find it doesn't run straight away on the Electron, you are likely to find yourself having to re-write the program considerably. It is best to bear in mind the Electron in the first place. Below are a number of features of the Electron which you must remember.

ROMS

Typing *HELP and REPORT reveals that the Electron uses BASIC II and OS 1.00. BASIC II means that you have several facilities including OSCLI for easy use of * commands. Although the Electron reports OS 1.00, it is what BBC owners would call OS 1.20. The Electron has never had any other OS, so of course as far as the Electron is concerned, it is OS 1.00. OS 1.00 means, for instance, that you can turn off the cursor using VDU23,1,0;0;0;0;, use characters from CHR$128, etc.

MODE 7

The Electron does not have MODE 7 at all – if you issue a MODE 7 command on the Electron you end up in MODE 6, which is the default mode of the computer. MODE 6 uses 8k of memory up instead of the 1k in MODE 7. If you are using MODE 7 for a header program that puts up a picture, the best thing you can do is to convert the MODE 7 into a MODE 4 screen using the modified version of our MODE 7 Screen Dump program from the last issue. If you are using MODE 7 for printing up instructions to a program or something similar, you are best to use MODE 1 which will allow you to use 4 colours. No modes on the Electron will allow you to alter one character on the screen by altering one byte in the memory.

SPEED

Most of the observations about speed were made in the Electron v BBC article elsewhere in this issue. A program that uses MODEs 4, 5 or 6 on the BBC will run on the Electron slightly slower. If the program uses MODE 3, a more noticeable delay will occur and if it uses MODEs 0, 1 or 2 it will be very much slower. For fast programs you should stick to MODEs 4 and 5. If you want to do anything fast in MODE 0, 1 or 2 you will have to use machine code.

KEYBOARD

The Electron keyboard has enough differences to make you think very carefully about which keys are suitable for controlling a program. For instance, a large key that has no set function on the BBC Micro is the TAB key – quite often this is used in programs to control various functions. The Electron, of course, does not have such a key. In short, the Electron does not have:

TAB
SHIFT LOCK

and keys which are in a different position/not on the BBC Micro are:

CAPS LOCK
COPY
[]
()
£_
^

;/
All user definable keys
FUNC

continued on page 22

using atari style joysticks on the bbc

There are two basic types of joystick available for computers – an analogue type and a digital type. If a computer has a socket which is designed purely for joysticks then they normally use the digital type. If the socket is to take other readings as well, such as on the Beeb, then the analogue type is used. The best type for games programs is a switch type as used by Atari, Commdore, Spectrum, etc. Good BBC joysticks are very hard to find, and those that are of a high quality are also of a high price. Also, on the whole, they are not that good for using with games. The answer, of course, is to interface an Atari style joystick to the BBC using the user port – this isn't all that easy, as not only does the actual interfacing require a lot of work, but also you will need a good machine code driver if you are to use the joystick with the commercially available programs.

MicroStyle have managed to interface a Spectravision Quickshot joystick to the BBC Micro – the Quickshot joystick is arguably the best joystick of its kind around. The cable to the joystick is approx. 1.25m long and ends in a connection that needs to be fitted onto the user port. The joystick itself consists of a helicopter style grip with a fire button on top and one on the side as well, depending on how you prefer using the joystick. The actual joystick itself is very responsive and feels comfortable.

The "heart" of the system is the software – this comes on cassette but can easily be transferred onto disk. Running the program is done by simply entering *JYSTK. A few pages of information on the joystick are displayed, followed by a menu. If you are playing an Acornsoft game that is on the menu, then you simply press that button and the joystick will be set up for that program – if it is not, then you have to enter the keys that the game you are about to load in uses. You define up, down, left and right and then you can decide on what fire commands you want. Most games you will only want to define one fire button, but for something like 'Rocket Raid', in which you can either fire missiles or drop bombs, you can define the fire button to do both. After deciding what keys the program uses, you can test to check everything works. Finally, control is returned to BASIC and a machine code driver is stored in &A00–&AFF.

The next thing to do is to load in the program – from there on you should be able to play the game using the MicroStyle joystick. The joystick will not work on any program that uses the memory between &A00 and &AFF but you should find that most programs will work (the main one I have found that it will not work with is 'Killer Gorilla').

The joystick will work only with programs that use INKEY with a minus value or *FX129 with a negative value for X.

The MicroStyle joystick works and performs well. At just under £20.00 it is reasonably priced and adds an extra dimension to most games. About the only pity is that they did not leave the 9 way D plug on the end of the joystick and make the lead from the user port end in a 9 way D socket. That way you could have used your own joystick if you have a preference. RECOMMENDED

We would like to thank MicroStyle for their help in this article. They can be found at 47 Cheap Street, Newbury. Tel: 0635 41929. Other branches at Daventry and Bath.

continued from page 21

Several programs use CAPS LOCK/CTRL to move the player left and right – on the Electron in the same position is CTRL/A. One program uses the function keys to select options and reads the keys using INKEY with a negative parameter. This cannot be done on the Electron. The FUNC/CAPS LOCK key can be picked up using INKEY-65.

SOUND

The Electron can only make one sound at a time. Therefore programs that on the Beeb utilise all the sound channels (which is most) will need some alteration to prevent a cacophony. You can, I am afraid, forget the &HFSCA,A,P,D format of sound. The channel number should be either 1 for tones or 2 for sound effects. The volume of the Electron's sound is either on or off – on is denoted by a negative number and off by 0. The pitch and duration are as on the BBC Micro. With the ENVELOPE command, only the first 8 numbers are significant – the last 6 numbers should be replaced by 126,0,0,–126,126,126 to make the Electron and BBC sounds the same.

TAPE FILING

Only the 1200 baud CFS is available on the Electron.

6845

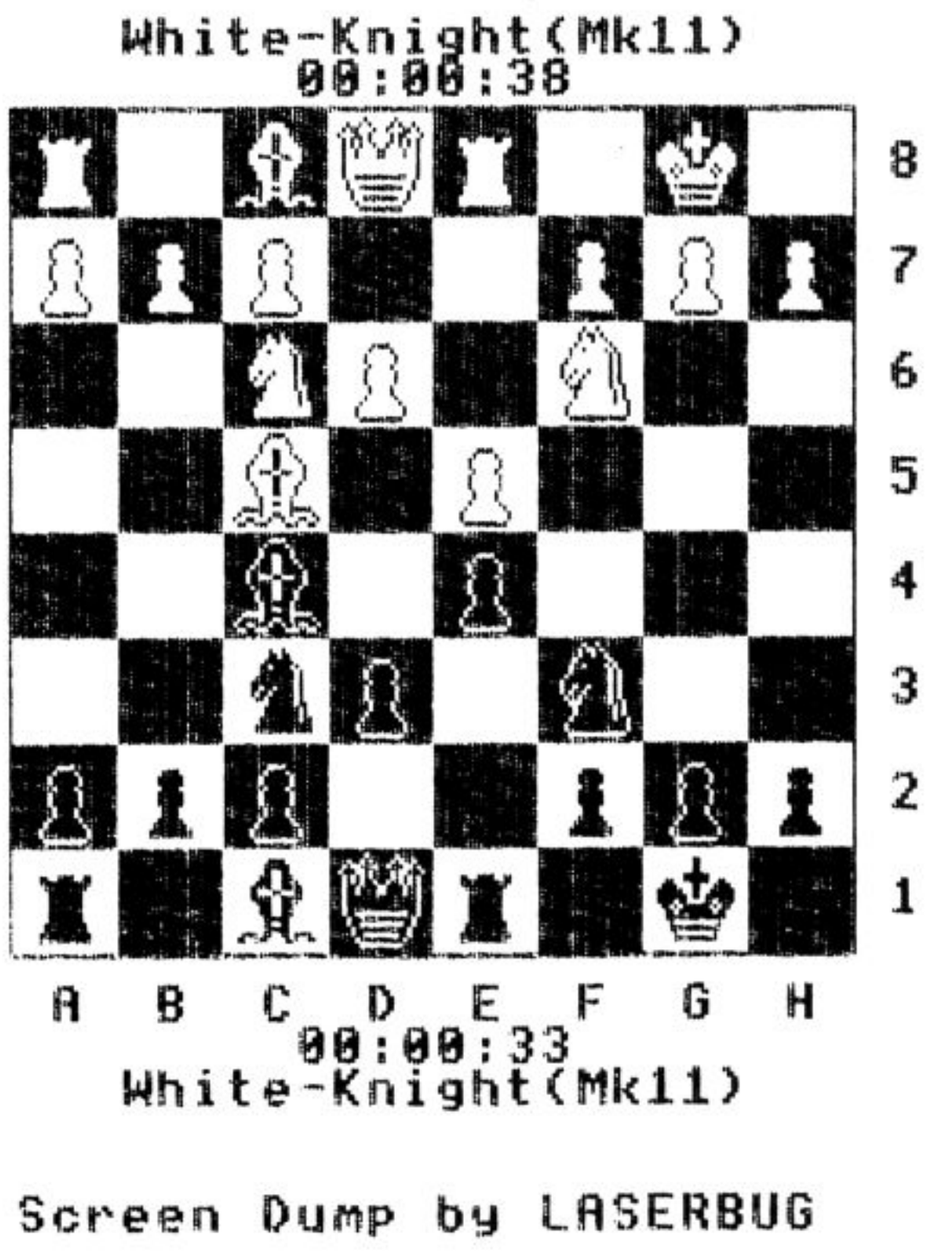
The Electron has no 6845 chip controlling the display – this means that several things are not possible. The first is that the *TV command does not work – on the majority of televisions and monitors it is necessary to do a *TV255 command to get the whole of the picture on the screen and so the top line of the display should be avoided. Many programs change the size and flash rate of the cursor – this is only done via a function in the 6845 and hence you must use the default cursor. Sideways scrolling is another facility carried out only by the 6845.

screen dump for white knight

'White Knight Mark II' by BBC Soft is a very good chess program. However, one of its biggest drawbacks is that it has no save game facility. White Knight is quite a complex and long program and so interfacing any user program with it proved very difficult. So instead we thought we would cheat a little and at least provide you with a way of getting a hard copy of the game. You can do this with virtually any program that doesn't use a *FX200 command to wipe out the memory on pressing BREAK.

When you get to the stage in the game where you want to leave it and print out the screen, press BREAK. Then save a section of memory with the command *SAVE SCREEN 5800+2800. Obviously it is much, much easier if you are using disks (White Knight II can easily be transferred onto disk). Then load in the following program and RUN it.

```
>L.
  10 REM SCREEN DUMP FOR
  20 REM WHITE KNIGHT MARK 11
  30 REM by Paul Barbour
  40 :
  50 REM MODE 4 Screen Dump
  60 REM by Jeremy Ruston
  70 :
  80 REM Version 1.0
  90 :
 100 REM (c) LASERBUG 1984
 110 :
 120    :::::
 130 :
 140 MODE4
 150 *LOAD SCREEN 5800
 160 PRINTTAB(0,28);SPC(160);TAB(0,0);
 170 PRINTTAB(30,7);SPC(10)
 180 PRINTTAB(30,8);SPC(9)
 190 PRINTTAB(30,9);SPC(9)
 200 PRINTTAB(1,28);"Screen Dump by LASERBUG"
 210 VDU2,1,27,1,65,1,8:FORL%=0TO39:VDU1,27,1,76,1,0,1
,2:FORT%=31TO0STEP-1:FORG%=7TO0STEP-1:A%=?(HIMEM+T%*320
+6%+L%*8):VDU1,A%,1,A%:NEXTG%,T%:VDU1,10:NEXTL%:VDU1,27
,1,50,1,12,3
>
```



The program only works on Epson printers (change the screen dump to one of your own if you have any other printer).

The method is very simple, MODE 4 (which is the mode White Knight works in) uses the memory between &5800 and &8000. MODE 7, which you are returned to on pressing BREAK, only uses the memory between &7000 and &8000. You would think that the computer clears the memory between &5800 and &8000 when it changes to MODE 7 – in fact it doesn't and only clears the area it needs. This means that the majority of the screen between &5800 and &7000 is intact. Saving this area means you still have a copy of most of the screen. You can then go into MODE 4 and load the screen back in – you will see a little bit at the bottom corrupted but the majority is intact. The program above clears the unnecessary parts and then prints it out. Simple!

There are numerous cassette recorders available suitable for use with home computers. However, most are music recorders that have been "brought into service" for use with computers. One of the few especially designed computer cassette recorders is the W.H. Smiths CPD8300 Computer Program Data Recorder.

The cassette recorder is much larger than a normal tape recorder measuring 10" (width) by 8" (depth) by 3" (height) approx. It is the normal "Smiths" brown and sits fairly neatly by the computer with all controls plainly in view. It is mains powered only – the battery powered cassette recorder is of more use to music players.

On the back of the recorder are FIVE inputs plus one switch! There is a load socket and a remote socket as usual. Then there are two save sockets – one for normal computers and one for computers like the Beeb that have a very load cassette output which can damage the mechanisms of some recorders. Also there is a "sound in" socket and a switch marked "Spectrum" and "Other". More on these later in the review.

On the top right hand corner is a fairly large 3 digit tape counter to keep track of your programs. The cassette door has a large window allowing you to see the contents of the tape and the window is also graduated to give an indication of how much tape is left.

Five piano type keys control the main operations of the recorder. In order these are SAVE (not record), LOAD (not play), REWIND/REVIEW, F.FWD/CUE and STOP/EJECT (which is larger than the others). To load a program you just press the load button! To save a program is slightly more complex – you have to press the load and save buttons together. This is a pity as "one touch record" is a fairly easy feature to build into the machine. When you are loading a program, pressing rewind or fast forward allows you to review or cue the tape.

In the middle of the recorder is a level meter, three LEDs, a speaker switch and a large pause button. The level meter is graduated between 0 top 10 and is also lit. It works on all sound – programs loading, programs saving and sound input. The three LEDs are load, save and tape run. The load LED is green and is normally the one constantly on. When the cassette motor is working at any time (i.e. loading or rewinding a program) a red tape run LED at the bottom flashes. When you save a program the green load light goes off and is replaced by a red save light.

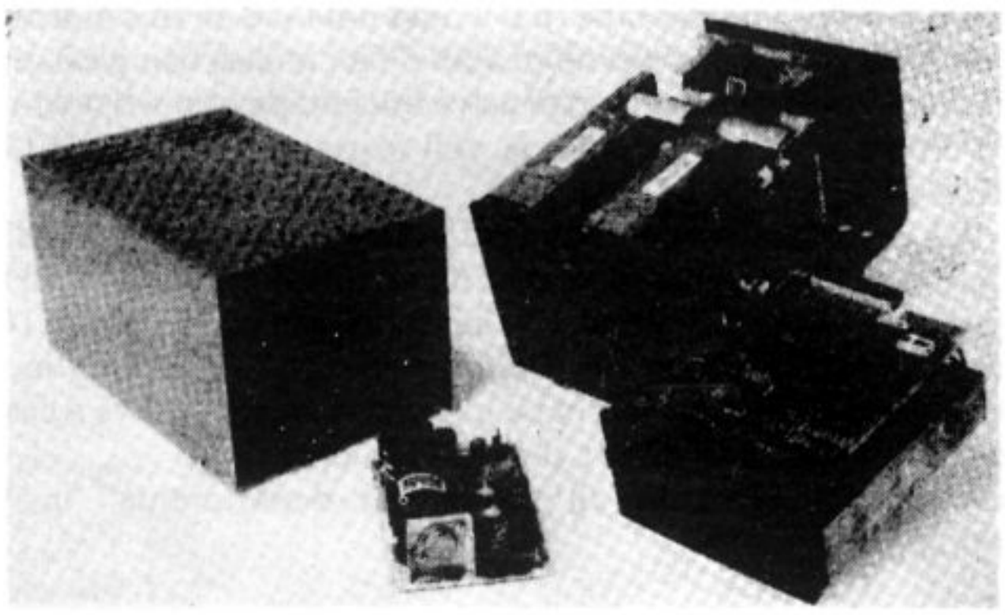
The speaker control has three positions – off, monitor and sound/amp. In the off position, the speaker is of course disconnected. In the monitor position, a quieter version of what is currently being loaded/recorded can be heard. The final position isn't of any direct use to BBC Micro owners. On the Spectrum which has the sound signal passed through both the load and save leads when not in use, it provides an immediate way of improving the sound considerably. On the BBC Micro, if you have a sound output lead fitted, then a simple lead terminating in a 3.5mm jack socket will provide you with a fairly good sound output (and much louder than the Beeb's own sound!).

Below the speaker control is a very large pause key. With the BBC Micro, assuming you have the correct cassette lead, the pause switch can be left down continuously, which will provide you with motor control. However, as the pause switch is electronic and not mechanical, the recorder will still be able to rewind and forward wind, just normal load or save operations are affected. If you wish to over-ride the BBC completely, then simply lifting the pause button will achieve this.

```
1490 COLOUR5
1500 PRINTTAB(0,19);channel2$
1510 PRINTTAB(0,20);min$;TAB(10,20);max$
1520 COLOUR6
1530 PRINTTAB(0,22);channel3$
1540 PRINTTAB(0,23);min$;TAB(10,23);max$
1550 COLOUR7
1560 PRINTTAB(0,25);channel4$
1570 PRINTTAB(0,26);min$;TAB(10,26);max$
1580 ENDPROC
```


from £175.

MICROWARE ANNOUNCE THE Z/ L RANGE OF FLOPPY DISC SUB SYSTEMS



The Control Data Family of Flexible Disk Drive is a single- or double-sided, random-access, low-cost storage device. Maximum storage capacity 1 megabyte on a 133.4-millimetre (5.25-inch) interchangeable disk. This disk drive is interchangeable with comparable products, providing industry compatibility in size, mechanical mountings, electrical interface, power requirements and physical appearance. CDC® FDD operates in single- or double-density formats. Single-density operation is achieved by using Frequency Modulation (FM) encoding, and in double-density operation uses Modified Frequency Modulation (MFM).

OPERATIONAL CHARACTERISTICS

	9408	9409	9409T
Capacity Unformatted			
Per disk	250.0 kbytes	500.0 kbytes	1 megabyte
Per track	3.1 kbytes	6.2 kbytes	6.2 kbytes
Formatted (16 sectors, 126/256 bytes)			
Per disk	163.84 kbytes	327.68 kbytes	655.36 kbytes
Per track	2.1 kbytes	4.1 kbytes	4.1 kbytes
Code	MFM	MFM	MFM
Transfer Rate	125 kbits/s	250 kbits/s	250 kbits/s
Average latency	less than 100 ms	less than 100 ms	less than 100 ms
Seek Time			
Track to track	less than 5 ms	less than 5 ms	less than 5 ms
Average Access	less than 80 ms	less than 80 ms	less than 132 ms
Setting time	less than 15 ms	less than 15 ms	less than 15 ms
Head Load Time (OPT)	less than 50 ms	less than 50 ms	less than 50 ms
Media	hard/soft sector	hard/soft sector	hard/soft sector

Rotational Speed	300 r/min	300 r/min
Track Density	48 TPI	96 TPI
Flux Reversal Density (track 39, side 1)	5876 FRI	5922 FRI
Number of Tracks	40	80

Inner recorded radius (side 0)	1.437 in (36.50 mm)	1.385 in (35.2 mm)
Outer recorded radius (side 0)	2.250 in (57.2 mm)	2.250 in (57.2 mm)
Inner recorded radius (side 1)	1.354 in (39.39 mm)	1.344 in (34.1 mm)
Outer recorded radius (side 1)	2.167 in (55.0 mm)	2.167 in (55.0 mm)

THE Z/ L RANGE

Type	Capacity	Tracks	No. of Drives
ZL141	250 k	40	1
ZL142	500 k	40	2
ZL241	500 K	80	1
ZL242	1 Mb	80	2
ZL291	1 Mb	160	1
ZL292	2 Mb	160	2

Microware Disk Drive Subsystems are plug compatible with the following: –
TANDY, VIDEO GENIE, BBC MODEL B, NABCOM, SUPER BRAIN, IBM/PC. and many more.

Microware also supply floppy drives, Winchesters and sub systems to OEM's and the trade at very aggressive prices, offering the best price performance ratio available. Call and ask for details.

CABINETS AND PSU
Cabinets of steel, aluminium or plastic are available in a choice of colours, and each unit is fully guaranteed.

AC 8151 40 WATT SERIES
The AC 8151 Switch-Mode Power Supply has been designed for use in small terminals and other similar equipment. The AC 8151 has dual line inputs and regulated outputs of: +5V at 2.5A, +12V at 2.0A, -12V at 0.1A. This compact unit features low magnetic radiation and is built to conform to International Safety and RFI Regulations.

Microware (London) Ltd.,
637a Holloway Road, London N19 5SS.
Telephone 272 6398/6237

40 80 switchable

On the far right of the cassette recorder is a power switch, level control and power LED. The level control affects everything – loading, saving, sound amp, etc.

The Smiths data recorder has proved extremely reliable in use – tapes which were previously unloadable now load fine – I have never found a cassette tape that would not load on this machine! It is extremely easy to use and in actual fact makes using cassettes a pleasure again. It is quite expensive but you pay for what you get – the Smiths cassette recorder is well worth the money. RECOMMENDED

W.H. Smith Computer Program Data Recorder (CPD8300). PRICE: £39.95 (when this cassette recorder is bought with a ZX Spectrum, it is discounted I believe by between £5 and £10. As Smiths now sell both the BBC Micro and the Electron, it is possible similar offers might be made? Smiths also have a cassette lead available suitable for the BBC Micro).
AVAILABILITY: At all "computer departments" in W.H. Smiths now.

*Since this review was written, we have had to take back our first cassette recorder because of a jammed level meter and a broken cassette door. Also another Data Recorder we have bought had its heads out of alignment and hence would not load all programs. The fault on the first cassette recorder we were assured was unusual (to the extent that we were virtually accused of breaking it ourselves!) and the heads can be adjusted manually with a very small screwdriver (although this should not be necessary).

verify

The most common way of verifying programs at the moment is to attempt to load them in over the ROM memory space. For instance, to verify the program ZAP you would use *LOAD ZAP 8000. This would attempt to load the program in the computer from location &8000 onwards. However, as this area is used for paged ROMs, in actual fact nothing happens, although the program seems to load in as usual. Hence the program can be verified.

However, a true verify routine as used on many computers does not just check to see the program will still load into the computer, but will check byte for byte that what was recorded was what is in the memory. Hence is you save:

```
10 REM This is a good program
```

alter the line to read

```
10 REM This is a poor program
```

a verify will show up that the recording has an error in it because the original one in memory (or what it thinks is the original one) is not the one on tape.

And so onto our program. The program will perform a complete byte by byte verify. It needs to be stored somewhere in memory out of the way, i.e. somewhere above your program. We found that in MODE7, the program sits quite happily at PAGE=&7900.

The way you should work is this. Save whatever program you are using, i.e. SAVE"ZAP". Then if our verify isn't already in memory, load in verify (PAGE=&7900:CHAIN"VERIFY" in MODE7 for instance). Loading in verify will have no effect on your original program (as long as your program isn't too long so that their is no room for verify and your program to fit in the computer at once). You will then be asked to enter the filename of the program (i.e. ZAP) and the value of PAGE for ZAP (i.e. &1900). ZAP will then be verified byte by byte and will give you the message O.K. if the program is fine or FAILED if it is not.

The program was originally written for a disk based BBC Micro. If you have a tape based BBC Micro or an Electron you should make the following changes:

```
100 IFLEN(file$)>  ORfiles$) ) )
110 . . .address <Return> for &E00:. . .
120 . . .THENmemory%=&E00:ELSE. . .
```

The program will work exactly the same – the only problem is that you must have motor control of your cassette recorder. You should now never have a BASIC program recorded badly if you check first. . .

```
>L.
  10 REM PROGRAM VERIFIER
  20 REM by Paul Barbour
  30 :
  40 REM Version 1.0
  50 :
  60 REM (c) LASERBUG 1984
  70 :
  80 CLS
  90 INPUT"Filename: "file$
 100 IFLEN(file$)>7OR file$=""THENPRINT'"Illegal filena
me":END
 110 INPUT"Start address <Return> for &1900: &"memory$
 120 IFmemory$=""THENmemory%=&1900:ELSEmemory%=EVAL("&
"+memory$)
 130 IFmemory%<00Rmemory%>&8000THENPRINT'"Illegal addr
ess":END
 140 start%=memory%
 150 file%=OPENIN(file$)
 160 REPEAT
 170   char%=BGET£file%
 180   IFchar%<>?memory%THEN250
 190   PRINTCHR$13;"&"~memory%-start%;
 200   memory%=memory%+1
 210   UNTILEOF£file%
220 CLOSE£file%
230 PRINT'"O.K."
240 END
250 CLOSE£file%
260 PRINT'"Failed"
270 END
```

music editor review

There are two ways to enter music onto a computer – one to play the music and the other to enter the notes one by one. Speaking as someone who enjoys playing the keyboard (keyboard meaning piano, synthesiser, organ, etc.) I find playing music on a computer keyboard a very hard task. Therefore, I prefer the second method.

Entering the notes by name is also a very tedious process when you are used to dealing with proper sheet music. 'Music Editor' by System Software overcomes all of these problems. The program we are reviewing here is the disk version, although both this and the tape version cost exactly the same!

When you start up, music editor asks (in MODE 4) if you want to load in a tune – answering Y allows you to load in a tune from disk (a tape/disk containing a number of tunes is available for a small charge separately). Next you are asked which key you want to work in. This means that unlike a program where you have to play the tune, you need not convert any notes. This is followed by a prompt to ask how many voices you are going to be using (1–3). You are then asked for the tempo (between 50 and 300 beats per minute in varying steps) and then if you would like to load in previously saved envelopes.

Continued on page 27

Microwriters cost quite a lot of money and so if you bought one (or two or three or four!) at the recent exhibitions, perhaps you might like to try incorporating it into your own programs for something other than a full keyboard. First of all you need to understand how the Microwriter interfaces to the BBC Micro.

The facts that (i) it connects up to the ADC port and (ii) four Microwriters can be connected at once to the BBC Micro should provide you with the answer. Each Microwriter is connected up to one of the ADVAL channels – not channel 0 which deals with only 2 switches but the normal channels 1–4. Keyboard one is linked to ADVAL (i) and so on.

So we now know how they connect – how can you get 6 switches on a channel designed for something completely different? To find out try typing in the following program:

```
>L.  10 REM Microwriter 1
     20 :
     30 MODE7
     40 VDU23,1,0;0;0;0;
     50 :
     60 REPEAT
     70   FORprint%=1TO4
     80     PRINTTAB(0,(10+(print%-1)));CHR$131;"Keyboard
  ";print%;":";CHR$134;FNmicrowriter(print%);SPC(1)
     90     NEXT
    110   UNTIL0
    120 :
    130 DEFFNmicrowriter(n%)=ADVAL(n%)DIV1024
  >
```

Assuming you only have keyboard one connected (and the other three connections are fitted with the dummy plugs) all four keyboards (or all four ADVAL channels) should show 0. If you press the right most button, you will find channel one shows 1. If you press the other buttons going anti-clockwise you will find that the others give 2, 4, 8, 16 and 32. If two keys are pressed together you will see the two numbers added up and so on (or if you are above the beginners level you get a 6-bit number with the bits arranged in the same order as the keys). To summarise the key values:

Right Hand Button – 1
Right – 2
Up – 4
Left – 8
Down – 16
Left Hand Button – 32

The easiest way to work the key values out is to number the keys from 5 to 0 (right to left) and then use 2 key number>.

Supposing you wanted to use the Microwriter as a kind of "Mouse" (i.e. a device to move a cursor around the screen such as is used with Apple's Lisa). Well, try the following program (apologies for taking the term mouse literally!).

```
 10 REM Microwriter 2
 20 :
 30 MODE4
 40 VDU23,1,0;0;0;0;
 50 :
 60 VDU23,128,0,0,0,30,61,255,0,0
 70 VDU23,129,0,0,0,120,188,255,0,0
 80 VDU23,130,56,84,124,124,124,56,16,16
 90 VDU23,131,16,16,56,124,124,124,84,56
100 VDU19,0,3,0,0,0
110 VDU19,1,4,0,0,0
120 :
130 X%=20:Y%=15:C%=128
140 :
150 REPEAT
160   VDU31,X%,Y%,C%
170   REPEAT
180     com%=FNmicrowriter(1)
190     UNTILcom%<>0
200   VDU31,X%,Y%,32
210   IFcom%=2THENX%=X%+1:C%=128
220   IFcom%=4THENY%=Y%-1:C%=130
230   IFcom%=8THENX%=X%-1:C%=129
240   IFcom%=16THENY%=Y%+1:C%=131
250   IFX%>39THENX%=39
260   IFX%<0THENX%=0
270   IFY%<0THENY%=0
280   IFY%>30THENY%=30
290   UNTIL0
300 END
310 :
320 DEFFNmicrowriter(n%)=ADVAL(n%)DIV1024
```

You can see that it is no harder to use the Microwriter to control a cursor than it is to use the BBC's keyboard. The advantage of the Microwriter is that it is very, very light to use and quite simple, especially with this kind of operation.

The alternative uses of the Microwriter are endless – I originally looked into this topic because I wanted a way of controlling a program remotely. My original thoughts were to wire up a single switch to the ADC port. Then I thought of the Microwriter, which of course has 6 switches. With that particular program I use the back arrow to go back a step in the program, the forward arrow to move on and the up arrow to start a part – this I can do from some distance away. Perhaps you use the BBC Micro to control some piece of hardware – maybe a robot arm such as Armdroid 1 from the user port. You could sit some distance away from the computer and control the robot from the Microwriter. Perhaps you use the BBC Micro to produce TV test signals. Going back to the keyboard is rather a nuisance to change a function, but what about using the Microwriter to control the program? The possibilities go on and on – and it all makes good use of the Microwriter.

The "Mouse" program only works when you are pressing one key – some applications will require you to be able to detect more than one key. The third and final program performs a complete scan and will tell you the status of all 6 keys.

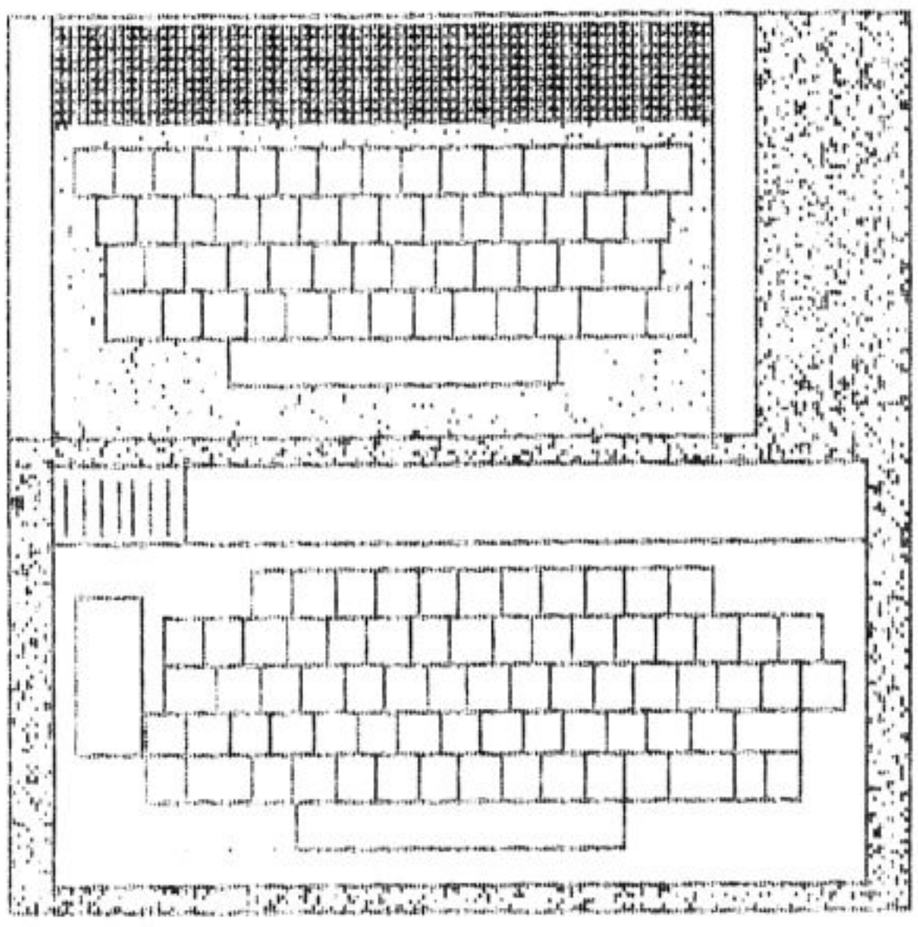
```
L.
   10 REM Microwriter 3
   20 :
   30 MODE7
   40 VDU23,1,0;0;0;0;
   50 :
   60 PRINTTAB(2);CHR$135;"RIGHT HAND BUTTON"
   70 PRINTTAB(2);CHR$135;"RIGHT"
   80 PRINTTAB(2);CHR$135;"UP"
   90 PRINTTAB(2);CHR$135;"LEFT"
  100 PRINTTAB(2);CHR$135;"DOWN"
  110 PRINTTAB(2);CHR$135;"LEFT HAND BUTTON"
  120 FORrow%=0TO5:PRINTTAB(22,row%);CHR$156:NEXT
  130 :
  140 REPEAT
  150   com%=FNmicrowriter(1)
  160   FORread%=5TO0STEP-1
  170     IFcom%DIV(2^read%)THENPRINTTAB(0,read%);CHR$1
29;CHR$157:com%=com%MOD(2^read%):ELSEPRINTTAB(0,read%);
" "
  180     NEXT
  190   UNTIL0
  200 :
  210 DEFFNmicrowriter(n%)=ADVAL(n%)DIV1024
>
```

The Acorn Electron is commonly described as a cut down BBC Micro with less interfacing capabilities but still using the same BASIC. How far is this true? What have Acorn got rid of to make the Electron £200 cheaper than the BBC Micro?

When you open the box of the Electron, which is packaged extremely well, you find:

The computer itself
A mains adaptor
An aerial lead
A User Guide
A book
An introductory cassette
Guarantee card

One thing that is surprising is just how much smaller the Electron is compared with the BBC Micro – the sketch below depicts the Electron placed on top of the BBC Micro in the top left hand corner:



The largest item inside the BBC Micro is the power supply as anyone who has ever opened the case will know. On the Electron one of the reasons that they have been able to keep the computer so small is that the power supply is separate. The normal method of having an external power supply to a computer is to have a box which sits by the computer with a mains lead coming out of one end and a power lead to the computer coming out of the other. Acorn have decided to incorporate the mains adaptor in the plug like the adaptors for children's electronic games – however, instead of being the same kind of size as those adaptors the Electron's power pack measures 630mm (w) × 870mm (h) × 550mm (d) and weighs 0.5kg! Despite its considerable size and weight, it does sit fairly happily in a power socket. The power lead coming from the power pack is a lengthy 3.25m – you can't call Acorn miserly. The power supply gives out 19V/14 Watts.

The power lead plugs in on the right of the computer – plugging it in switches the Electron on and unplugging it switches off the computer. It is a great, great pity that Acorn decided to use this instead of adding a switch on the side. As they should have learned from other manufacturers, this kind of arrangement means the socket has a very finite lifetime.

On the left side of the computer there are four sockets – one is for the cassette and is identical in format to the BBC Micro in all respects. There are three sockets for connecting up a display as on the BBC – one for a television, one for a B&W monitor (although this uses a normal UHF out type socket rather than the BNC connector used on the Beeb) and one for an RGB monitor.

The only other connection to the outside world is the Electron's edge connector – this again unfortunately is of a similar type to the ZX Spectrum's and has proved to have a limited lifetime if used frequently. The logic behind using such a connection I would imagine is that the only thing you are meant to plug into there is Acorn's own expansion box which is designed to be screwed into place and left on the computer permanently.

The "Welcome" tape for the Electron (called the Introductor Cassette) has a number of programs on it, most of which are re-written versions of their BBC counterparts. Most programs (because the authors have had so much more time to get to grips with the computer) are superior to their BBC counterparts. The programs are:

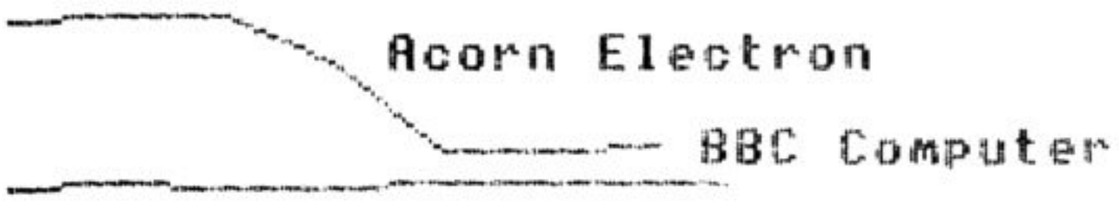
KEYBOARD: A keyboard trainer which has much better presentation than the BBC version.
SKETCH: A much more comprehensive drawing program than the one on the BBC tape which was no more than a simple "Etch-a-Sketch". The facilities even include a fill command.
PIANO: Rather than just bars of colour on the screen when you play notes, the notes are actually printed on the staves.
DODGEMS: Is the same as the Beeb version.
BIORHYTHMS: Fairly similar to the Beeb version except of course that MODE 7 is not used.
CLOCK: Similar to the Beeb version.
GOMOKU: As included on one of the BBC Soft tapes?
PATTERNS: Same as the Beeb version except that you press number keys instead of function keys.
MARSLANDER: Simple lunar lander game.
BUGZAP: Very very simple space invader type game.
ISLAND: As on creative graphics. The interesting point to note here is that it would take so long for the computer to actually draw the picture they *LOAD in the screen.
PLANETS: Similar in principle to ISLAND but of course with a "spacey" picture.

The programs are recorded on both sides of the tape – you can easily get rather confused on side B as you have to skip past a few programs recorded for use with the book. A few more instructions informing you of this would have helped!

The user guide with the Electron is 292 pages long compared with the Beeb's user guide of 520 pages – the Electron user guide has a fairly large section on assembler but the BASIC keyword index with at least one page on each command is not followed. However, to supplement the Electron User Guide, a 138 page book entitled Start Programming With The Electron by Masoud Yazdani is included. This is a very good book indeed and is well suited to introducing the whole topic of computing and BASIC.

The keyboard of the Electron is quite different from the BBC Micro's. One additional key marked FUNC is on the Electron but the TAB and SHIFT LOCK key on the BBC Micro is gone and several other keys are incorporated in others. The BBC Micro has 74 keys compared to the Electron's 56. The red f0–f9 keys of the BBC Micro are incorporated into the Electrons number keys – to get the user defineable functions you have to press FUNC at the same time (it is used like a SHIFT key). CAPS LOCK is also on the FUNC key and to alter this you have to use SHIFT – a small orange light next to FUNC indicates whether you are locked into capitals. All the letter keys and a few symbol keys have BASIC words underneath them – pressing FUNC and one of these keys will get the word. This is not true single key entry but is some of the way there. Five keys double as editing keys and produce two normal characters. The editing functions are the cursor keys and the COPY key – the COPY key has been moved up to the upper right hand corner just below BREAK (!). Above the editing keys are two characters – one on the left and one on the right. To get the one on the left you have to press SHIFT and to get the one on the right use CTRL. Not an ideal situation but certainly one that you can live with. As far as the "feel" of the keys go, the Electron's keyboard is much more positive than the Beeb's.

The BASIC on the Electron is the same as the BBC's (BASIC II) but the big crunch comes in the speed the Electron operates. The BBC Micro operates at the same speed no matter what mode it is in – the Electron's speed varies greatly. We carried out a test on all modes on the BBC and Electron which consisted of a simple FOR loop – no use as such was made of the screen. The differences in speed were considerable:



Although the BBC worked the same in all modes, in MODEs 0, 1 and 2 the Electron worked considerably slower. The situation wasn't so bad in MODE 3 and in MODES 4, 5 and 6 it wasn't too much different from the BBC. This means that although any program written on the BBC Micro in the lower MODEs will work without little speed difference on the Electron. If they use the higher modes they are likely to be unusable if they depend to any extent on the high speed of the BBC Micro. These conclusions were backed up by further tests.

In MODE 2 we ran a simple program printing triangles:

```
MODE 5
======
BBC COMPUTER    995
ACORN ELECTRON  1359
```

but when we tried it in MODE 5, the Electron ran almost as fast as the Beeb:

```
MODE 2
======
BBC COMPUTER    1263
ACORN ELECTRON  3525
```

ran with a corresponding result in MODE 4:

```
MODE 4
======
BBC COMPUTER    181
ACORN ELECTRON  261
```

A similar test printing lines in MODE 0:

```
MODE 0
======
BBC COMPUTER    278
ACORN ELECTRON  709
```

Another big difference between the Beeb and the Electron is the SOUND – the Electron can only sound one channel at once and the Envelope command is only a cut down version of the Beebs.

In my opinion, the Acorn Electron is a good computer. A surprisingly large number of people are buying the BBC Micro as their first ever computer – a £400 investment in a new hobby which you never know if you will like is rather stupid. A more sensible idea is to try a computer such as the Electron. The main "market" that can be seen by Acorn is school children who use the BBC Micro at school and so will want an Electron at home. The Electron is very good for beginners – the manual and book that comes with it is especially geared for complete novices. The support is already available for it – in software, books and magazines. Hardware support is starting slowly and will grow considerably in the next year. The Electron has a very good future and is set to seriously affect the sales for other computers in the under £200 price bracket, i.e. Sinclair, Commodore, Oric, etc.

Continued from page 24

The Treble and Bass staves then appear on the screen. The facilities you then have at your disposal for editing are considerable. The best way to use the program is with a score in front of you. To start off with you should pick the required volume or envelope – 16 envelopes are already defined for you, although these can be altered. You can pick the duration of the note by using two keys and where the note is to be placed on the staves with the cursor keys. The note can then be stored either as a note (dependant on the key signature), a sharp, a flat, a natural or of course a rest. Bar lines can be added where necessary and passages repeated. Editing functions include the ability to insert and delete notes and to play back any part of the tune so far. Everything can be altered as required, meaning the whole thing is quite flexible. Each voice is edited separately. When you have built up the entire piece, you can then play it by returning to the command menu.

This menu has the options to play the piece, save the tune, edit a voice, alter the tempo or the envelopes, load or save the envelopes or restart the program. The manual is quite helpful and even includes a program to use tunes created with the editor in your own program.

Music Editor is a very flexible and well written program indeed. It is excellent at converting any musical score onto the BBC Micro. It interfaces directly with a Graphical Envelope Generator, allowing you to shape the sound better. I have been very impressed with the program indeed and would recommend it to anyone interested in music. It is a pity that System are not using this program as the main one in their current advertising campaign.

'Music Editor' costs £9.00 inclusive for either the tape or the disk version. It is good value for money when compared with other similar products currently on the market. Let's hope they manage to interface a real keyboard to the Beeb and see what they can do then. . . RECOMMENDED

TOP QUALITY PRINTERS!
ROCK BOTTOM PRICES!

EPSON RX80	£248 + VAT
EPSON RX80F/T	£268 + VAT
EPSON FX80	£358 + VAT
EPSON MX100	£398 + VAT
EPSON FX100	£458 + VAT
SHINWA CP80	£248 + VAT
STAR 510	£248 + VAT
STAR 515	£348 + VAT

We also have large stocks of sundries including ribbons, fanfold and roll paper, continuous labels, dust covers, etc.

For FREE BROCHURE and DISCOUNT ORDER FORM

write now to:

Datatech Limited
8 Bellingham Close, Bury,
Lancs., BL8 2TU

credits

© LASERBUG 1984. Reasonable care is taken to avoid errors in this magazine. However, no liability is accepted for any mistakes which occur. The opinions expressed in this magazine are those of the authors of each individual article and do not necessarily represent those of the Editor. No material may be reproduced in any way whatsoever (for any reason) without the written consent of the Editor.

LASERBUG is edited by Paul Barbour

LASERBUG is a User Group for the BBC Microcomputer and Acorn Electron affiliated to the ACC.

All correspondence should be sent to LASERBUG, 10 Dawley Ride, Colnbrook, Slough, Berks., SL3 0QH. Presetel Mailbox Number 91 999 1005. Prestel page 8008128. Response frames on 80081289.

Advertising rates are available on request.

Code	Description	Price & Postage
SAMPLE COPIES		
SC01	Sample copy of LASERBUG	1.00 (H)
MEMBERSHIPS +		
MS01	UK (and Forces) 12 Month Membership (12 Issues)	14.00 (A)
MS02	UK (and Forces) 6 Month Membership (6 Issues)	7.00 (A)
MS03	Worldwide (European and Outside Europe) 12 Month Membership (12 Issues)	18.00 (A)
MS04	Postal Band A (see below) Air Mail 12 Month Membership (12 Issues)	22.00 (A)
MS05	Postal Band B (see below) Air Mail 12 Month Membership (12 Issues)	24.00 (A)
MS06	Postal Band C (see below) Air Mail 12 Month Membership (12 Issues)	26.00 (A)
BACK COPIES ++		
BC01	Reprint of LASERBUG Issue 1 (16 pages)	1.60 (B)
BC02	Backcopy of LASERBUG Issue 2 (16 pages)	1.25 (B)
BC03	Backcopy of LASERBUG Issue 3 (24 pages)	1.25 (B)
BC04	Backcopy of LASERBUG Issue 4 (20 pages)	1.25 (B)
BC05	Backcopy of LASERBUG Issue 5 (32 pages)	1.25 (B)
BC06	Backcopy of LASERBUG Issue 6 (24 pages)	1.25 (B)
BC07	Backcopy of LASERBUG Issue 7 (24 pages)	1.25 (B)
BC08	Backcopy of LASERBUG Issue 8 (24 pages)	1.25 (B)
BC09	Backcopy of LASERBUG Issue 9 (24 pages)	1.25 (B)
BC10	Backcopy of LASERBUG Issue 10 (24 pages)	1.25 (B)
BC11	Backcopy of LASERBUG Issue 11 (28 pages)	1.25 (B)
BC12	Backcopy of LASERBUG Issue 12 (24 pages)	1.25 (B)
BC13	Backcopy of LASERBUG Issue 13 (28 pages)	1.25 (B)
BC14	Backcopy of LASERBUG Issue 14 (28 pages)	1.25 (B)
BC15	Backcopy of LASERBUG Issue 15 (28 pages)	1.25 (B)
BOOKS		
IN01	Let Your BBC Micro Teach You To Program by Tim Hartnell	5.50 (C)
IN02	The BBC Micro Revealed by Jeremy Ruston	6.50 (C)
IN03	36 Challenging Games for the BBC Micro by Tim D. Rogers and Chris Callender	5.00 (D)
BINDERS		
BD01	LASERBUG Binder suitable for 12 Issues	5.00 (E)
PRINTERS +++		
PR01	Star STX-80 (RRP £160) B	150.00 (F)
PR02	Star Gemini 10X (RRP £287) A	260.00 (F)
PR03	Star Gemini 15X (RRP £413) A	360.00 (F)
PR04	Star Delta 10 (RRP £459) A	400.00 (F)
PR05	Star Delta 15 (RRP £620) A	530.00 (F)
PR06	Star Radix 10 (RRP £654) A	560.00 (F)
PR07	Star Radix 15 (RRP £792) A	670.00 (F)
PR08	Shinwa CTI CP80 Type I (RRP £229) A	220.00 (F)
PR09	Olivetti JP101 (RRP £229) C	210.00 (F)
PR10	MCP 40 (RRP £149) D	140.00 (F)
PR11	Juki 6100 (RRP £516) F	420.00 (F)
PR12	Canon PW1080A (RRP £367) A	320.00 (F)
PR13	Canon PW1156A (RRP £459) A	400.00 (F)
PR14	Canon PJ1080A (RRP £498) G	440.00 (F)
PR15	Riteman (RRP £286) A	260.00 (F)
PR16	Seikosha GP-100A Mark II (RRP £247) A	220.00 (F)
PR17	Seikosha GP-250A (RRP £288) A	270.00 (F)
PR18	Seikosha GP-700 (RRP £489) I	430.00 (F)
PR19	Epson RX80 (RRP £321) A	290.00 (F)
PR20	Epson RX80F/T (RRP £367) A	330.00 (F)
PR21	Epson FX80 (RRP £489) A	450.00 (F)
PR22	Epson FX100 (RRP £654)A	570.00 (F)
RGB MONITORS ++++		
MN01	Sanyo SCM 14N (RRP £229) A	200.00 (F)
MN02	Sanyo SCM 14M (RRP £401) B	310.00 (F)
MN03	Sanyo SCM 14H (RRP £574) C	440.00 (F)
MN04	Fidelity CM14 (RRP £229) D	220.00 (F)
3" DISKS – Available from March '84 +++++		
DK01	Hitachi HFD305 3" Disk Drive (RRP £252)	230.00 (F)
DK02	Pack of ten 3" disks (RRP £80.50)	70.00 (I)
CABLES		
CB01	Printer cable for any of the above	10.00 (G)
CB02	Monitor cable for any of the above	10.00 (G)
CB03	7 pin DIN – DIN and Remote Jack	2.50 (B)
CB04	7 pin DIN – 3 Jacks	2.50 (B)

SOFTWARE – Coming soon. . .

Postage rates on page 9

CODE	DESCRIPTION	PRICE	P&P
	SUB TOTAL		
		TOTAL	

NAME.....

ADDRESS.....

..... MEMBERSHIP NUMBER..... DATE.....

ALL ITEMS NORMALLY EX STOCK PLEASE ALLOW 7 DAYS FOR DELIVERY *

* Allow 4–6 weeks for 1.2 ROMs

THIS ORDER FORM SUPERCEDES ALL OTHERS

If you run out of room or do not want to cut up your LASERBUG you may order on a separate piece of paper or take a photocopy of this form.